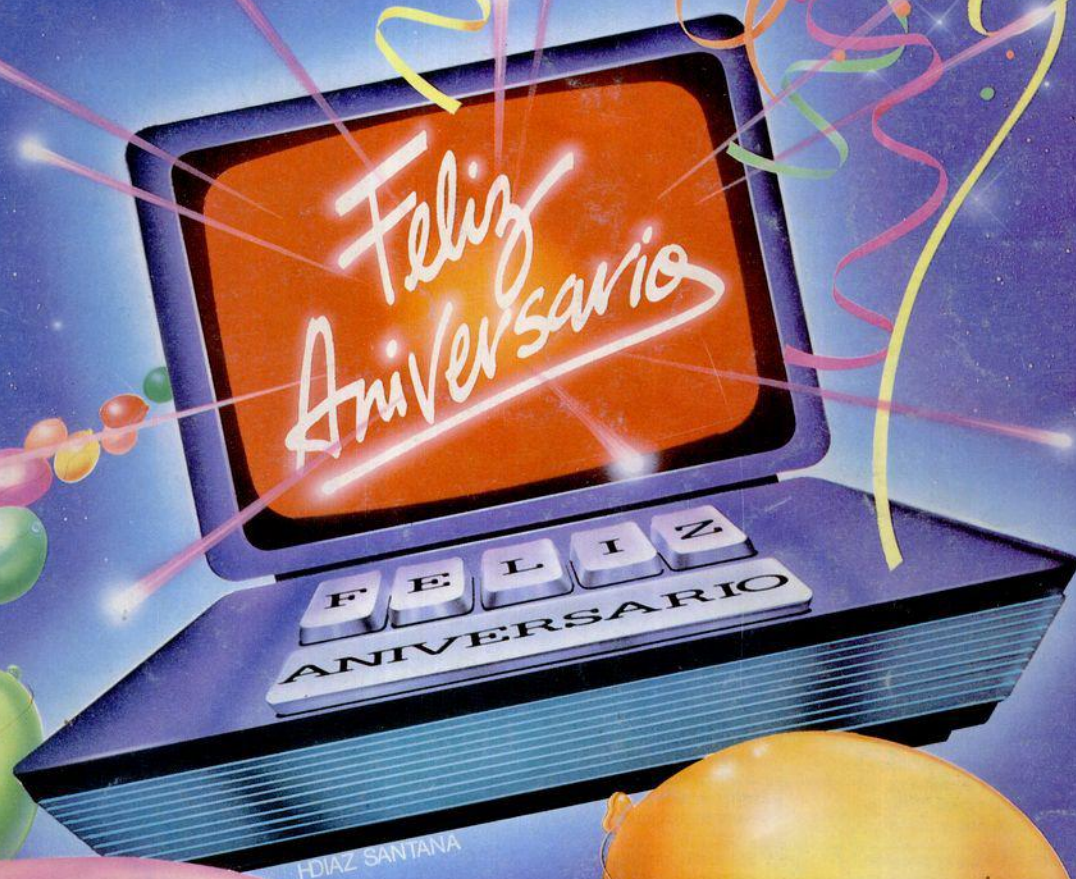


commodore

WORLD

Nº 23 - FEBRERO 1986

350 PTAS.



¡¡Cumplimos dos años!!

- Construye tu propio joystick
- Listados perfectos
- Reset paso a paso



# RITEMAN C+:

## BUILT-IN COMMODORE INTERFACE



**Novedad:**  
**C+Versión NLQ**

**INFORUNNER**

# RITEMAN C+:

### ESPECIFICACIONES

1. **Método de impresión**  
Serie de impactos por matriz de 9x9
2. **Cabezal impresor**  
9 agujas (reemplazable) con expectativa de vida de 50 millones de caracteres a 100 % de uso continuo.
3. **Velocidad de impresión**  
Impresión a 105 cps uni o bi-direccional, búsqueda lógica optimizada, capacidad de impresión 45 lpm a 10 cpi., 200 ms de salto de línea con espaciado de 1/6".
4. **Espaciado interlínea**  
1/6", 1/8", 7/72", programable N/72", N/216".
5. **Alimentación de papel**
  - a) Tractor ajustable entre 4" y 10".
  - b) Alimentación a fricción para hojas sueltas entre 4" y 9.5".
6. **Corte de papel**  
Se puede imprimir desde principio de página hasta una pulgada de fin de papel.
7. **Selección de caracteres**  
96 caracteres ASCII, 96 itálicos, 82 caracteres gráficos Commodore. Definición de caracteres por usuario a través de software fácilmente.
8. **Gráficos**  
Normal 480 puntos por línea. Doble densidad 960.
9. **Modos de impresión**

*Modo Commodore*  
Standard, reverse (negativo), mayúsculas y minúsculas y caracteres gráficos.

*Modo plus*  
Standard, doble impresión, enfatizado, itálica, supra y subíndices mitad altura.
10. **Tamaños de impresión**

*Modo Commodore*

  - a) Normal
  - b) Expandido
  - c) Expandido doble
  - d) Expandido triple
  - e) Expandido cuádruple.

*Modo plus*

  - a) Normal (10 cpi), 2,1 mm (a) x 2,55 mm (h).
  - b) Normal expandido (5 cpi)
  - c) Comprimido (17 cpi)
  - d) Comprimido-expandido (8,5 cpi).
11. **Test escritura**  
Manteniendo apretado LF (salto de línea) cuando se acciona el interruptor de puesta en marcha ON.
12. **Subrayado continuo.**
13. **Tabulador**  
Existen 32 posiciones de tabulación por línea.
14. **Longitud de formato**  
El standard de 66 líneas, con espaciado de líneas y longitud del formato programable. VFU eléctrico.
15. **Tamaño buffer**  
1 línea standard.
16. **Diagnóstico de detección de errores en modo Commodore.**
17. **Posicionador de papel continuo**  
Desplazable parte anterior o posterior.
18. **Interface**  
Directo a Commodore salida RS232 a través de conector DIN (el cable se suministra con la impresora).
19. **Grosor papel**  
Máximo 2 copias, 40 g/m<sup>2</sup> peso continuo, sensor papel.
20. **Operación con interruptores**  
De fácil acceso.
21. **Caracteres por línea**

Normal: 80

Normal expandido: 40

Comprimido: 132

Comprimido expandido: 66
22. **Cinta impresora**  
Autorretintada, mediante tarpon interno, cartucho compacto de cinta negra fácilmente reemplazable y bajo coste.
23. **Tamaño**  
115 (h) x 275 (f) x 405 (a).
24. **Tensión alimentación**  
220 v/ 50 Hz.
25. **Entrada de papel**  
Frontal-horizontal.
26. **Retroceso**  
Por software.
27. **Instrucciones de formateado de tablas alfanuméricas**
28. **Elevadores inferiores standard**  
(para ubicación del papel continuo debajo de la impresora).
29. **Motor: paso a paso**
30. **Indicadores:**
  - Conexión general
  - Fin de papel
  - On line
31. **Interruptores**  
ON/OFF, On line (SEL), salto de página (FF), salto de línea (LF).
32. **Compatible con la mayor parte de software existente (incluso Commodore).**

**DATAMON**  
DATAMON, S. A.

REPRESENTACION EN ESPAÑA DE

**RITEMAN**

-IMPRESORAS PROFESIONALES-

- PERIFERICOS ORDENADORES
- IMPRESORAS Y MONITORES
- CONEXIONES Y ACCESORIOS



Consejo de Dirección:  
Neil D. Kelley

Adjunto a la Dirección:  
Eugenio S. Ballesteros

# Commodore WORLD

Commodore World  
está publicado por SIMSA  
y la colaboración  
de todos nuestros lectores.

Coordinadora:  
Valerie Shanks

Redacción/Dpto. Técnico:  
Diego Romero,  
Alvaro Ibáñez,  
José Manuel Fernández

Colaborador:  
José Luis Errazquin

Diseño:  
Miguel Angel Hermosell

Secretaría de dirección:  
Lola Hermosell

Distribución y Suscripciones:  
Fernando Rodríguez (dirección),  
Angel Rodríguez,  
Juan Márquez (suscripciones)  
Tel.: 221 86 71/77

SIMSA  
Coordinadora Valerie Shanks  
c/ Barquillo, 21-3º Izqda.  
28004 Madrid. Tels. (91) 231 23 88/95  
Télex: 45522 CCBE E

DELEGACION EN BARCELONA:  
Sant Gervasi de Cassoles, 39  
despacho 4  
08022 Barcelona  
Tels. (93) 212 73 45/212 88 48

C.I.F. A.-28-609758

Distribuidora  
SGEL  
Avda. Valdelaparra, s/n.  
Polg. Ind. de Alcobendas  
Madrid

Distribuidor en Sudamérica  
A/C de Guatemala  
17 Calle, 13-72, Zona 11  
Tel. 480402

GUATEMALA, C.A.  
LIBRERIA HACHETTE, S.A.  
Rivadavia, 739  
1002 Buenos Aires  
Tel. 34-8481 al 85

Solicitado control O.J.D.

PROHIBIDA LA REPRODUCCION  
TOTAL O PARCIAL DE LOS  
ORIGINALES DE ESTA REVISTA  
SIN AUTORIZACION HECHA POR  
ESCRITO

NO NOS HACEMOS  
RESPONSABLES DE LAS  
OPINIONES EMITIDAS POR  
NUESTROS COLABORADORES

Imprime:

**B**IBERDOS S.A.

Germán Pérez Carrasco, 24.  
28027 Madrid  
Depósito Legal: M-2944-1984

## SUMARIO

<b>4</b>	<b>SUPERINTERESANTISIMO</b>	<b>42</b>	<b>MARKETCLUB</b>
<b>7</b>	<b>CONSTRUYE TU JOYSTICK</b>	<b>44</b>	<b>COLABORACIONES</b> • Master music • Glosario • Cuestionario • Sopa de letras • Cuentas corrientes
<b>10</b>	<b>LISTADOS PERFECTOS</b>		
<b>14</b>	<b>RESET: INICIALIZACION PASO A PASO</b>	<b>55</b>	<b>DIRECTORIO</b>
<b>20</b>	<b>HELICOPTERO DE RESCATE</b>	<b>56</b>	<b>CARTA BLANCA Y SEAMOS PREGUNTONES</b>
<b>24</b>	<b>BASIC 4.5</b>	<b>59</b>	<b>RINCON DEL CODIGO MAQUINA</b>
<b>32</b>	<b>DIBUJOS DINAMICOS</b>	<b>65</b>	<b>COMENTARIOS COMMODORE</b> • Platine 64 • Hardball • Prontuario para el C-64 • Revs • Diccionario para el C-64 • Juegos y programas
<b>40</b>	<b>MAGIA</b>		

## P ROXIMO NUMERO

¿Sabes utilizar 64 sprites a la vez?

Ampliación del Basic para el VIC-20

MEMOCO - El brazo mecánico

• ... y todas vuestras colaboraciones



Commodore World es miembro de CW Communications/Inc., el grupo editorial más grande y de mayor prestigio del mundo en lo que se refiere al ámbito informático. Dicho grupo tiene a cargo la edición de más de 50 publicaciones relacionadas con los ordenadores en 20 países. Nueve millones de personas leen una o más publicaciones del grupo todos los meses. El grupo editorial está integrado por: ALEMANIA: Computerwoche, Infowelt, PC Welt, Computer Business, Run (Commodore). ARGENTINA: Computerworld/Argentina. ASIA: Asian Computerworld. AUSTRALIA: Australia Computerworld, Macworld and Directories, PC World. BRASIL: Data News, PC Mundo. DINAMARCA: Computerworld/Danmark, PC World y Run (Commodore). ESPAÑA: Computerworld/España, Commodore World, PC World. ESTADOS UNIDOS: Computerworld, InCider, InfoWorld, PC World, 80-Micro, Mac World, Micro Market World, 73 magazine, Run (Commodore), Focus Publications, Amiga World, On Communications. FINLANDIA: Mikro. FRANCIA: Le Monde Informatique, Golden (Apple), OPC (IBM) y Distributique, Theoreme. HOLANDA: Computerworld Benelux, PC World. ITALIA: Computerworld Italia, PC Magazine. JAPON: Computerworld Japan. MEXICO: Computerworld/-México. NORUEGA: Computerworld Norge, PC Mikrodota. REINO UNIDO DE GRAN BRETAÑA: Pc Business World, Computer News, Computer Business. REPUBLICA POPULAR DE CHINA: China Computerworld. Computerworld, China Computerworld Monthly. SINGAPUR: Asian Computerworld. SUECIA: ComputerSweden. MikroDatorn. Svenska PC. SUIZA: Computerworld Schweiz. VENEZUELA: Computerworld Venezuela.

## ¡¡Feliz cumpleaños Commodore World!!

**E**n este mundillo de la informática donde las cosas aparecen y desaparecen a una velocidad que marea al personal, es de suponer que el distinguido público agradece un punto de apoyo donde agarrarse. En su segundo cumpleaños, Commodore World tiene el derecho de sentirse un poquito orgulloso de encontrarse no solamente presente en el mercado, sino más fuerte que nunca. Dos años en la vida de una revista son muchos años. Suponen mucho esfuerzo, muchos artículos, muchos

listados, algunas meteduras de pata (todo hay que decirlo), y sobre todo mucho compañerismo y comunicación entre nosotros, los currantes, y vosotros que, al fin y al cabo, sois nuestra razón de ser. Juntos hemos compartido muchas experiencias. Hemos viajado a los Estados Unidos, y otro suscriptor nuestro está a punto de pasar unos días en Europa. Organizamos el primer concurso nacional de programación a nivel de colegios en un intento de fomentar el interés de los chavales, concienciar a los profesores y para que los centros tuvieran la oportunidad de

ampliar, o en algunos casos comenzar, su aula informática, gracias a los premios donados por Microelectrónica y Control. Nuestra última promoción ha sido el sorteo de 8 Commodore 128 que han supuesto un agradable regalo de Reyes para los afortunados.

Commodore World siempre ha tenido como meta el promocionar los equipos Commodore porque creemos que es la mejor línea de ordenadores dentro de su sector que ha aparecido en el mercado. Son máquinas con muchas facetas e innumerables prestaciones y

## Corrección de la reseña de ARKOFOTO

Le debemos una disculpa a la casa Arkofoto de Paseo de Gracia, 22, 2º 1.ª, de Barcelona, de cuyos productos sacamos una reseña en el

número de enero de Commodore World. Al hablar del joystick Yanjen, distribuido en España por Arkofoto, sin querer le cargamos todos los impuestos habidos y por haber encima del precio de venta que se quedó en la suma nada despreciable de 43.000 pesetas en vez de 2.990 pesetas (más el IVA).



*Joystick Yanjen  
Distribuido por Arkofoto  
Precio: 2.990 ptas. (más el IVA)*

## ¡ATENCIÓN!

Mirad con cuidado el nuevo programa para teclear los listados en la página 10. Os quitará muchos dolores de cabeza.



**commodore**  
**WORLD**  
Nº 23 - FEBRERO 1986 - 350 PTA.

*Feliz Aniversario*

**¡¡Cumplimos dos años!!**

- Construye tu propio joystick
- Listados perfectos
- Reset paso a paso

nuestra intención es que los lectores de Commodore World utilicen la revista para sacar el mayor rendimiento posible de sus ordenadores. Muchos de vosotros habéis ido creciendo junto con nuestra publicación, lo que supone para nosotros una enorme satisfacción.

Al iniciar nuestro tercer año de vida sólo nos queda agradecer el apoyo y constancia que nos habeis demostrado, y dejar constancia de nuestro compromiso tanto con los más expertos como con todas aquellas personas que empiezan ahora a introducirse en el mundo de los "Commodorianos". Todos podeis contar siempre con nuestra ayuda y dedicación para poder disfrutar de este mundo apasionante que es la informática.

## CAMBIAMOS LAS BASES DE LAS COLABORACIONES

A partir de este mes vamos a cambiar las bases de las colaboraciones que nos enviáis, suprimiendo el sistema del sorteo celebrado cada seis meses. A partir de este número, el autor de la mejor colaboración del mes recibirá la cantidad de 5.000 pesetas. Elegiremos el mejor truco de "Magia" cada mes, que será premiado con 1.000 pesetas.

También estamos dispuestos a publicar artículos que sean trabajos con una buena redacción y documentación y, sobre todo, que sean ORIGINALES. Obviamente los autores de estos trabajos recibirán un pago por su artículo publicado.

**Avisamos  
a todos los suscriptores  
que nuestro  
departamento de  
suscripciones ha  
cambiado de teléfonos:**

(91)

**221 86 71**

**221 86 77**

## Bajada de precios de los ordenadores

Buenas noticias para nuestro país en cuanto a los precios de los ordenadores de importación. Microelectrónica y Control, los representantes oficiales de la casa Commodore en España, han anunciado una reducción en su lista de precios que se debe a dos factores: el arancel específico de 15.000 pesetas aplicable a los ordenadores de 64 Ks ha dejado de tener efecto a partir del 1 de enero y, en segundo lugar, la desaparición de los gravámenes interiores y del impuesto de tráfico de empresas, ambos sustituidos por el IVA.

Los nuevos precios son los que presentamos a continuación:

Commodore 128 = 71.300 ptas.

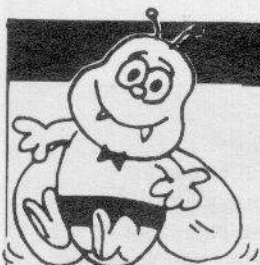
Commodore 64 = 44.500 ptas.

Unidad de disco 1541 = 49.900 ptas.

Commodore PC = 353.000 pesetas.

(Estos precios no incluyen el IVA)





## METEDURAS DE PATA

- Los errores del Basic 4.5 ya han sido localizados. Los tenéis al final del artículo que corresponde a este número (la tercera parte). Para colmo, la "chapucilla" que hicimos para que la tecla RESTORE no dejara colgado al ordenador también estaba mal. Olvidaros de ella porque está incluida en las correcciones del artículo.
- Para variar, en la respuesta a una carta en la página 39 del número 22, aparece  $2\ 16 + \text{FRE } (0)$ . Parece ser que la "flecha arriba" entre el 2 y el 16 ha salido disparada de nuevo.

## TIRADA DE COMMODORE WORLD Certificación de 28.000 ejemplares

Según certificado firmado por la auditoría internacional DELOITTE HASHINS SELLS, S. A., el 5 de marzo de 1984,

la tirada de Commodore World es de 28.000 ejemplares.

Este certificado se encuentra en las oficinas de S.I.M.S.A. para comprobación de cualquier persona o entidad que lo desee.

## Clave para interpretar los listados

**T**odos los listados que se publican en esta revista ha sido ejecutados en el modelo correspondiente de la gama de ordenadores COMMODORE. Para facilitar la edición de los mismos en la Revista y para mejorar su legibilidad por parte del usuario, se les ha sometido a ciertas modificaciones mediante un programa escrito especialmente para ello. Para los programas destinados a los ordenadores VIC-20 y COMMODORE 64, en los que se usan frecuentemente las posibilidades gráficas que aparecen del teclado, se han sustituido los símbolos gráficos que aparecen normalmente en los listados por una serie de letras entre corchetes [] que indican la secuencia de teclas que se deben pulsar para obtener el carácter deseado. A continuación se da una tabla para aclarar la interpretación de las indicaciones entre corchetes:

[CRSRD] = Tecla cursor hacia abajo (sin SHIFT)

[CRSRU] = Tecla cursor hacia arriba (con SHIFT)

[CRSRR] = Tecla cursor a la derecha (sin SHIFT)

[CRSRL] = Tecla cursor a la izquierda (con SHIFT)

[HOME] = Tecla CLR/HOME (sin SHIFT)

[CLR] = Tecla CLR/HOME (con SHIFT)

Las indicaciones [BLK] a [YEL] corresponden a la pulsación de las teclas de 1 a 8 junto a la tecla CTRL. Lo mismo sucede con [RVSON] y [RVSOFF] respecto a la tecla CTRL y las teclas 9 y 10.

El resto de las indicaciones constan de la parte COMM o SHIFT seguidas de una letra, número o símbolo —por ejemplo [COMM+] o [SHIFTA]—. Esto indica que para obtener el gráfico necesario en el programa deben pulsarse simultáneamente las teclas COMMODORE (la que lleva el logotipo) o una de SHIFT y la tecla indicada por la letra, el número o el símbolo, en el ejemplo anterior: COMMODORE y + o SHIFT y A, respectivamente.

En los signos gráficos además se cuenta el número de veces que aparece. Por ejemplo, [7 CRSRR] equivale a 7 pulsaciones de la tecla cursor a la derecha y [3 SPC] tres pulsaciones de la barra espaciadora.



# Construye tu joystick

Todos los que no tengan joystick, o aunque ya lo tengan quieren otro, ahora tienen la oportunidad de autoconstruirse uno. Por supuesto, a un precio mínimo.

**M**i joystick "murió" mientras un amigo y yo jugábamos un partido donde One-on-One. En un contraataque sorpresa se oyó un Crack! y el jugador se quedó parado. Bueno, no del todo. Podía moverse hacia los lados y hacia atrás, pero no hacia adelante. Inmediatamente desmonté el joystick e intenté arreglarlo. Imposible. La pieza de la palanca que hacía contacto en el circuito impreso estaba rota. No tenía arreglo. Cualquier intento de pegarla, sujetarla, etc., sólo habría servido para que se rompiera a los dos o tres días. Era una pena. El joystick no es que fuera de los buenos, pero en su tiempo me había costado 4.500 pesetas. No me duró ni un año.

Decidí no volver a comprar otro Joystick como aquél. Mirando por las tiendas puede ver otros modelos, mejores y más baratos, pero no quise comprarme ninguno. Sabía que de mucho

utilizarlos acaban por romperse. Entonces decidí construirme uno por mí mismo.

## Cómo construir un joystick

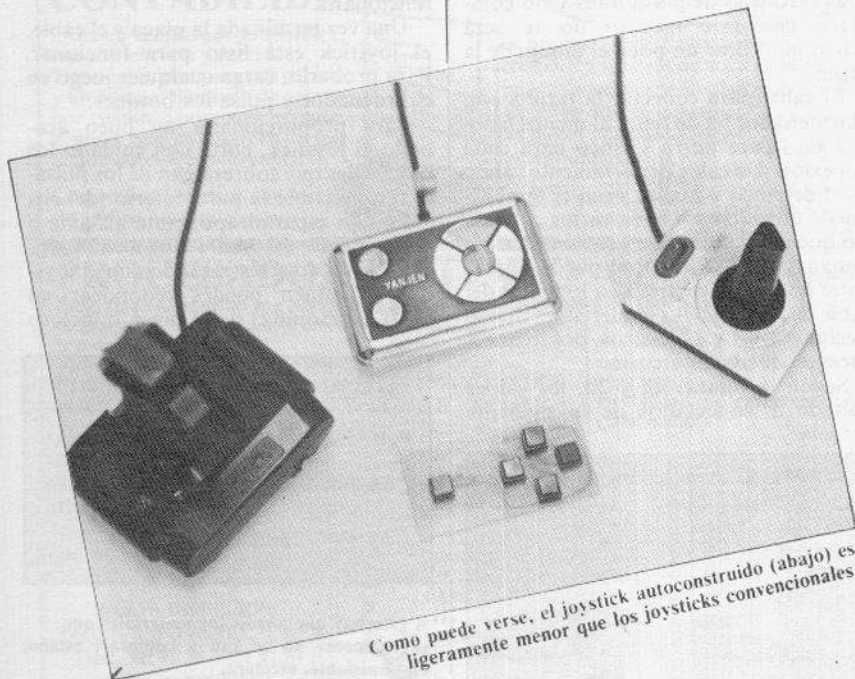
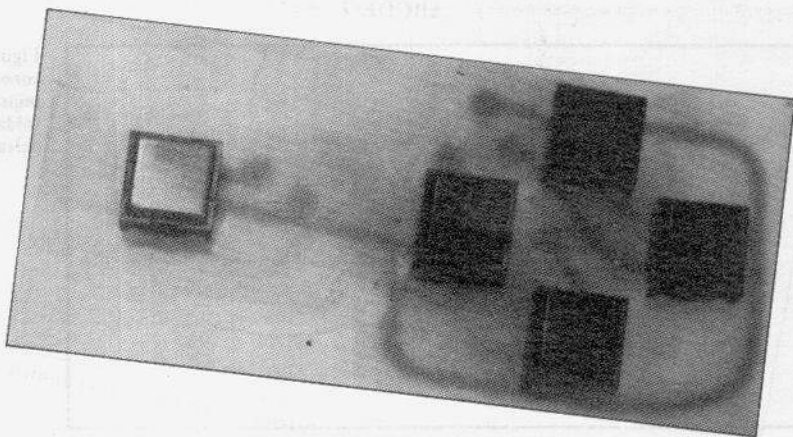
Es muy difícil construir un joystick de los tradicionales, es decir, de palanca. Las partes mecánicas no son muchas, pero no se encuentran en ningún lado. La mejor solución es construirse algo cómo un "joy-card", es decir, un joystick "plano" con botones.

Lo único que se necesita es un soporte, que puede ser una placa de circuito impreso, cinco pulsadores, cable y un conector de nueve patillas para enchufarlo al ordenador. En este artículo encontraréis una tabla con los precios aproximados de estos materiales.

Cualquiera que haya abierto un joystick habrá visto que el funcionamiento no tiene ningún misterio. Todo lo que hace es establecer contactos entre el polo negativo (tierra) del ordenador y los cinco terminales que se utilizan para señalar las cuatro direcciones y el disparo. En un apéndice de la Guía del usuario tenéis el esquema de las conexiones de los ports de juegos con la descripción de cada patilla. También está reproducido en la figura 1 de este artículo.

Los contactos señalados como JOY0 a JOY3 (pines 1 a 4) corresponden a las direcciones arriba, abajo, izquierda y derecha, respectivamente. El contacto 6 corresponde al botón de disparo. Este contacto también se utiliza con el lápiz óptico.

Para indicarle al ordenador un movimiento en cualquier dirección lo único que hay que hacer es establecer un contacto entre el Pin 8 (tierra) y cualquiera de los demás. Por poner un ejemplo: si con un cable uniéramos el Pin 8 con el Pin 1 se produciría el mismo efecto que al mover el joystick hacia arriba. Las diagonales se obtienen pulsando simultáneamente dos botones adyacentes. En un joystick convencional no es posible moverse a la vez a la izquierda y a la derecha, pero con este joystick sí. Esto



Como puede verse, el joystick autoconstruido (abajo) es ligeramente menor que los joysticks convencionales.



ABCDE F

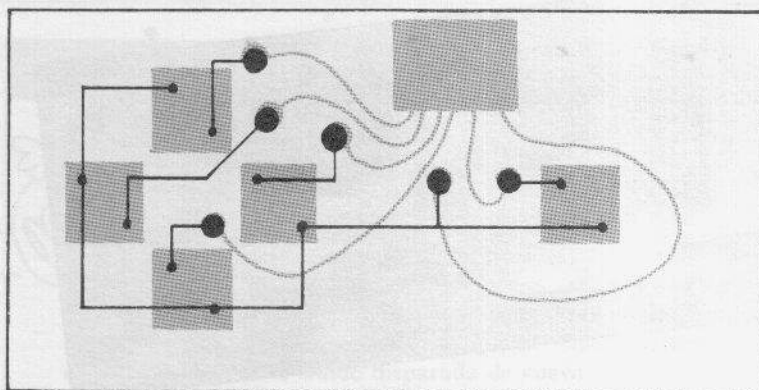


Figura 2a.—Placa para el joystick con un conector. En negro las pistas del circuito impreso (vistas desde el lado de las soldaduras). En gris la colocación de los pulsadores y los cables de conexión.

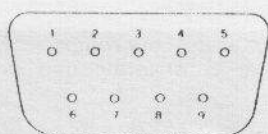


Figura 1.—Los Ports de juego. En el C-64 los dos son casi iguales, con la excepción de que el lápiz óptico sólo puede conectarse en el Port I.

Contacto n.º	TIPO	NOTA
1	JOY 0	
2	JOY 1	
3	JOY 2	
4	JOY 3	
5	Potenciómetro Y	
6	Lápiz-óptico	
7	+ 5V	MAX. 100mA
8	Tierra	
9	Potenciómetro X	

puede causar problemas en algunos juegos, por ejemplo en Realm of Impossibility. No he encontrado ningún otro juego en el que el joystick dé problemas.

En la figura 2 tenéis dibujado a tamaño natural dos placas para el circuito impreso del Joystick. En esencia son las dos iguales, la única diferencia es que una (la segunda) está preparada para que los cables sean soldados directamente a la placa. La primera deja un hueco libre para colocar un conector y que de esta forma el "joy-card" sea independiente del cable. Esta forma de conexión es más limpia, pero también más cara, pues hay que comprar los conectores, que además son difíciles de encontrar.

Si eres zurdo, o no te gusta tener el botón de disparo a la izquierda y los controles a la derecha, sólo tienes que invertir el dibujo de la placa.

Los pulsadores, que se pueden encontrar en cualquier tienda de electrónica, tendréis que escogerlos como más os gusten, porque hay una gran variedad. Los que yo utilizo son cuadrados y no "suenan" demasiado cuando se pulsan.

Estar oyendo "click!, click!", mientras se está jugando no es demasiado agradable.

Una buena idea es comprar cuatro de un color y uno rojo para el botón de fuego. Los botones cuadrados suelen tener las patillas de conexión en diagonal (como están en el dibujo). Si compras otro tipo de pulsadores y no coinciden con este modelo no te será difícil modificar un poco el dibujo de la placa.

El cable para conectar la tarjeta con el ordenador ha de tener al menos 6 polos: uno para tierra y cinco para cada conexión. Un cable perfectamente válido es el de cinco y malla, aunque también puede utilizarse cable de varios colores. Lo que más conviene es que sea flexible y manejable. La longitud puede oscilar entre uno y tres metros. La longitud del cable de los joysticks comerciales tiende a estar entre 1 y 1,5 metros, por lo que a veces se queda algo corto.

Según las figuras 2a y 2b, los cables han de ir conectados de la siguiente manera:

Joystick	Port juegos
A	1
B	4
C	3
D	2
E	6
F	8

ABCDE F

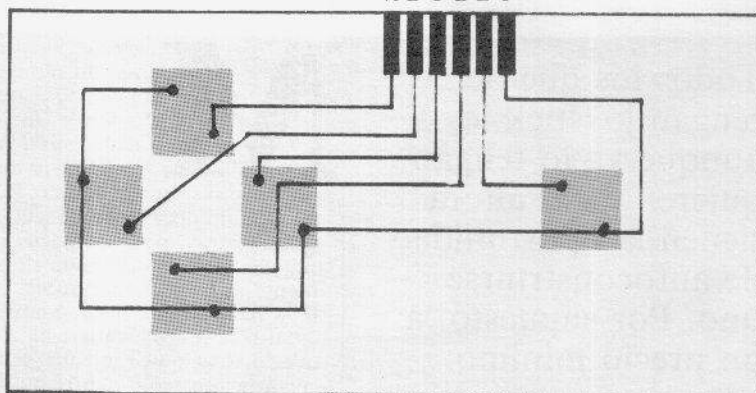


Figura 2b.—Placa para el joystick sin conector. Los cables han de ir soldados en los seis contactos que hay en la parte superior.

Si has elegido la opción 1 verás cómo en el conector que compres (vuelvo a repetir que no es muy barato, aunque también se puede conseguir desgastando algún aparato de radio viejo o algo parecido) hay una separación entre E y F. Esto es para que no introduzcas al revés el conector, ya que entonces no funcionaría.

Una vez terminada la placa y el cable, el joystick está listo para funcionar. Para probarlo, carga cualquier juego en el ordenador y pulsa los botones.

Para proporcionarle un buen acabado al joystick, corta con cuidado los terminales que sobresalgan de los pulsadores y recubre la parte inferior del circuito con esparadrapo, cinta aislante o algo similar. Esto hará que no te "pinches" cuando estés jugando con el joystick. También puedes redondear un poco las esquinas de la placa o incluso

#### Tabla de precios (aproximados)

Pulsadores (uno):	35-100 ptas.
Placa:	150-200 ptas.
Conector 9 pines:	200-300 ptas.
Cable:	60-80 ptas./metro.
Conector 6 pines (Opcional):	400-600 ptas.

A esto hay que añadir los materiales que seguramente no se van a comprar: estaño, cinta aislante, etcétera.

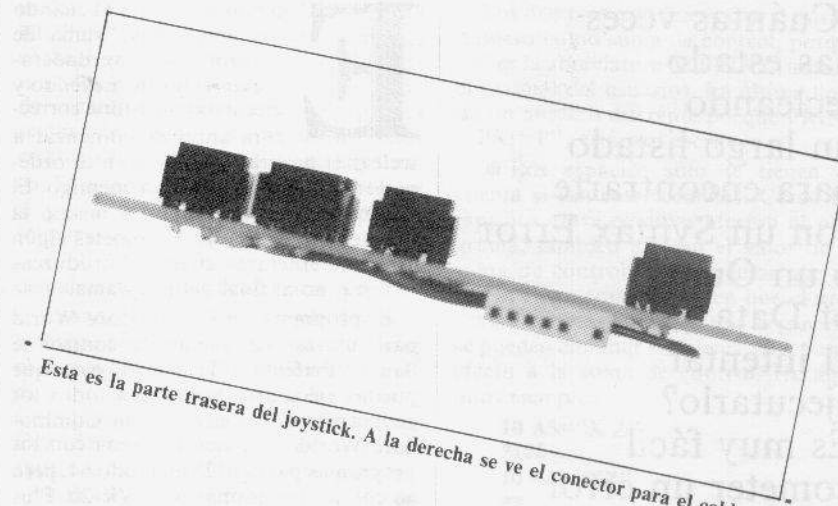


meterlo todo dentro de una cajita. Si eliges esto último procura que no sea demasiado grande ni profunda, porque no podrías agarrarlo con comodidad.

Si alguno no tiene muchos conocimientos de electrónica puede perfectamente colocar todos los botones en una cajita metálica u otro soporte cualquiera y realizar las conexiones. Estas conexiones son:

**H**ay que tener cuidado al conectar los cables al joystick. Si por casualidad las cosas se mueven hacia la izquierda cuando deberían hacerlo hacia la derecha, repasa las conexiones.

1. Un cable que una un contacto de cada uno de los pulsadores. Este cable debe ir al pin número 8 del port de juegos.



Esta es la parte trasera del joystick. A la derecha se ve el conector para el cable.

2. Un cable que vaya desde cada pulsador al pin del port de juegos correspondiente (son 1=arriba, 2=abajo, 3=izquierda, 4=derecha y 6=disparo). Eso es todo lo que hay que hacer.

Este joystick funciona con cualquier ordenador que tenga conexiones estándar para joystick, es decir, Commodore 64, 128, Vic-20... Pero no con el C-16, ya que los conectores para los ports de

juegos son "exclusivos" de la casa Commodore y no se encuentran en las tiendas (¿en qué estarían pensando cuando diseñaron el C-16?). Una solución (que sería más arriesgada, porque invalida la garantía) es abrir el C-16, soldar unos cuantos cables en los contactos de los conectores y sacar dos clavijas estándar por la parte trasera del ordenador. Que lo intente quien no quiera un "joystick Commodore".

## Para que su COMMODORE trabaje

**casa de  
software**

### CONTABILIDAD-64

Líder en ventas, por su sencillez, rapidez, eficacia y precio. Tiene capacidad para 600 cuentas y un número ilimitado de apuntes por cuanto el programa permite generar nuevos discos en los que continuar el ejercicio contable.

Contabilidad-64 es un producto de software autosuficiente



PARA QUE SU LIQUIDACION DE I.V.A. NO LE SORPRENDA

que permite tener en todo momento acceso a los ficheros de manera que puedan modificarse los datos contenidos en ellos, aún cuando estos ya hayan sido validados. Esta posibilidad da una total libertad al usuario en el manejo de la información.

#### OFERTA

Hasta el 30 de Marzo  
Versión B con I.V.A. y  
Programa I.V.A. sólo.  
Precio con I.V.A.  
INCLUIDO.

### I.V.A. - 64

- Listado de facturas recibidas
- Listado de facturas emitidas
- Informe contable (I.V.A. deudor y I.V.A. acreedor)
- Diario

CONTABILIDAD 64 A -	23.000,- Pts.
CONTABILIDAD 64 B -	25.000,- Pts.
CONTABILIDAD 64 + I.V.A. -	31.000,- Pts.
PROGRAMA I.V.A.	7.000,- Pts.
CONTABILIDAD PROFESIONAL VERSION CASSETTE - 7.900,- Pts.	

FUNCIONA SOLO  
CON EL CARTUCHO  
DE CONTABILIDAD

### PROCESADOR DE TEXTO

Programa en cartucho con posibilidad de grabación de documentos en cassette o diskette.

Caracteres castellanos y catalanes tanto en pantalla como en impresora. Posibilidad de utilizar todo el set de caracteres de la impresora. Márgenes, numeración de páginas, encabezamientos, pies de página, etc.

#### PROCESADOR DE TEXTO

Los tres acentos y la diéresis se obtienen pulsando F1, F2, F3 o F4 y a continuación la vocal correspondiente como en una máquina de escribir convencional.

Posibilidad de cartas personalizadas (mail merge).

P.V.P. 14.900,- pts.

DIGANOS QUE IMPRESORA USA.  
TENEMOS EL PROGRAMA QUE NECESITA.

#### VERSIONES PARA:

- SEIKOSHA SP 800
- IBM Compatibles (STAR, EPSON...)
- MPS 801 y compatibles COMMODORE
- ASCII Serie o paralelo (RITEMAN C +, STAR...)

Adquiéralos en cualquier establecimiento autorizado o directamente a:

Casa de Software, s.a.

NUEVA DIRECCION:

TAQUIGRAFO SERRA, 7, 5.º B

Tels. 321 96 36 - 321 97 58

08029 BARCELONA

REALIZAMOS  
DEMOSTRACIONES  
DE NUESTROS PROGRAMAS  
PARA EL PUBLICO.  
¡¡ LLAMENOS !!  
Solicite catálogo GRATUITO  
de nuestros productos.

☐ Deseo recibir información de los siguientes programas:

☐ Deseo recibir contra reembolso los siguientes programas:

Nombre: .....

Dirección: .....

Población: .....



¿Cuántas veces has estado tecleando un largo listado para encontrarte con un Syntax Error o un Out of Data al intentar ejecutarlo? Es muy fácil cometer un error mientras estás tecleando el listado de un programa, pero no es tan fácil encontrarlo después. Algunas veces, la línea que falla no es la que contiene el error. Esto es bastante desconcertante, sobre todo para los principiantes.

**E**n esas ocasiones es cuando los programas "suma de control" son verdaderamente valiosos. Leyendo y ejecutando la rutina correctora antes de comenzar a teclear el programa tendrás en tu ordenador a un amigo, no a un enemigo. El ordenador chequeará por sí mismo la línea que introduzcas. Si cometes algún error, te enterarás cuando introduzcas la línea, no al final del programa.

El programa de Commodore World para utilizar las sumas de control se llama "Perfecto". Te servirá para que puedas ejecutar a la primera todos los programas que aparezcan en Commodore World. También funciona con los programas para el 128 en modo 64, pero no con los programas para Vic-20, Plus 4 o C-16.

Te habrás fijado en que este mes, todos los listados de Commodore World llevan un ".XXX" en el margen derecho de cada listado: **No teclees nunca ese número en el listado**, tan sólo te servirá para compararlo con el que aparezca en la pantalla. Si el número que aparece cuando introduces la línea en el ordenador coincide con el que está en la línea del listado, puedes estar seguro de que lo que has tecleado es correcto. Si los dos números no coinciden, comprueba la línea que acabas de introducir y corrige el error.

#### Cómo usar a "Perfecto"

Teclea el listado que aparece al final de este artículo y guárdalo en cinta o en disco antes de ejecutarlo. Si cometes algún error cuando introduzcas los datos, un mensaje aparecerá en la pantalla avisándote. Haz las correcciones oportunas y salva la nueva versión. También él incluye su propia suma de control, para que cuando funcione puedas hacer pruebas.

Cuando quieras teclear algún programa de los que aparecen en Commo-

dore World, carga y ejecuta el programa "Perfecto". Dos números SYS aparecerán en la pantalla: apúntalos y ténlos a mano. Son los SYS para conectarlo y desconectarlo.

Puede que por alguna razón necesites desactivar a "Perfecto" cuando estés introduciendo el listado. El único modo de hacerlo sin desconectar el ordenador es tecleando SYS 40794 y pulsando RETURN. Pulsar STOP/RESTORE no desconectará a "Perfecto".

Después de haberlo conectado, introduce las líneas del programa como haces normalmente. La única diferencia que notarás es que cuando introduzcas una línea pulsando la tecla RETURN, aparecerá en la pantalla un número de uno, dos o tres dígitos debajo del cursor. Este número, llamado suma de control, estará siempre entre 0 y 255.

Si el número coincide con el que aparece listado en la revista, la línea es correcta y puedes teclear la siguiente línea de programa encima del valor de la suma de control. Si el número es diferente, busca en la línea errores de escritura u omisiones. (Mira la sección titulada "Corrigiendo líneas"). Haz los cambios necesarios y pulsa la tecla RETURN de nuevo para introducir la línea. Sigue así hasta que hayas tecleado todo el programa.

Cuando hayas terminado, desconecta el corrector (tecleando el SYS que apareció en la pantalla cuando lo conectaste) y graba el programa de manera normal. Si ejecutas el programa y te aparece un error Out Of Memory, apaga el ordenador y vuélvelo a encender. Esto borrará definitivamente a "Perfecto" de la memoria del 64.

Puedes salvar parte de un programa para seguirlo tecleando después. Si has apagado el ordenador tendrás que volver a cargar el programa "Perfecto", ejecutarlo y a continuación cargar el programa con el que estabas trabajando. Listalo y continúa donde lo dejaste.

# LISTADOS



● EL PROGRAMA SE GUARDA EN LA ZONA ALTA DE LA MEMORIA, NO OCUPA APENAS ESPACIO Y PERMITE LA EDICION NORMAL DE CUALQUIER PROGRAMA BASIC.

● LAS PALABRAS CLAVE PUEDEN SER ABREVIADAS Y LOS ESPACIOS ENTRE SENTENCIAS NO SE TIENEN EN CUENTA. LOS TEXTOS QUE VAYAN ENTRE COMILLAS HAN DE SER TECLEADOS TAL Y COMO APARECEN EN EL LISTADO.

● ES CASI IMPOSIBLE QUE UNA LINEA INCORRECTA DE UN VALOR CORRECTO COMO SUMA DE CONTROL. "PERFECTO" SE DARA CUENTA INCLUSO DE DOS CARACTERES QUE ESTEN CAMBIADOS DE ORDEN.

● SI TODO HA IDO BIEN MIENTRAS TECLEABAS, CUANDO HAGAS RUN EL PROGRAMA FUNCIONARA A LA PRIMERA.

#### Cómo aparecen los listados

Cómo ya dijimos anteriormente, los listados que aparezcan en Commodore World (que a partir de ahora irán todos a 35 columnas exactamente) contendrán siempre un ".XXX" al final de cada línea. Es el valor de la suma de control y no debe ser tecleado. Si tecleas estos valores, el programa no funcionará. Además, nunca obtendrás la suma de control correcta. Por ejemplo, si en el listado aparece:

```
10 PRINT X .50
```

debes teclear:

```
10 PRINT X
```

como suma de control aparecerá:

```
* 50
```

El punto que aparece en el listado indica que el valor de la suma de control

es 50. Después de que pulses la tecla RETURN, se calculará una suma de control para la línea que acabes de introducir. En este ejemplo, si la suma de control que aparece es \*50, puedes seguir tecleando la siguiente línea encima del \*50 que apareció en la pantalla. El asterisco causará un Syntax Error si pulsas RETURN por equivocación mientras el cursor está sobre la suma de control. Esto sirve para prevenir que accidentalmente puedas borrar una letra.

Cuando estés introduciendo los programas, puedes encontrar fácil utilizar el cambio mayúsculas/minúsculas. Pulsa simultáneamente las teclas Shift y Commodore para pasar a minúsculas. Cuando teclees caracteres gráficos con shift, aparecerán como letras mayúsculas.

Utiliza la tecla shift sólo cuando sea necesaria, por ejemplo con [SHIFT L] o [PI]. Si necesitas utilizar abreviaturas te será mucho más fácil leerlas en modo minúsculas. Ten en cuenta que la abreviatura para PRINT# es P [SHIFT R] y no ?#. ?#no es lo mismo que PRINT# o P [SHIFT R]. Si haces la sustitución, el valor de la suma de control no será el mismo y el programa no funcionará, aunque al listar el programa veas que la línea está aparentemente bien.

#### ¿Qué es lo que "Perfecto" chequea?

Lo que viene a continuación son las características de este programa:

● La línea se chequea sólo después de haber sido interpretada y convertida a tokens por el Basic. Esto hace que PRINT y ? den la misma suma de control. Si alguna línea demasiado larga necesita abreviaturas, la suma de control seguirá funcionando igual de bien. Si te equivocas al teclear alguna palabra clave (o pones alguna abreviatura incorrecta) el valor de la suma de control cambiará. He aquí algunos ejemplos:

```
10 ?X * 50
```

```
10 PRINT X * 50
```

```
10 PRINP X * 124
```

Los dos primeras líneas dan el mismo número como suma de control, porque "?" es la abreviatura de PRINT (mira en el manual del usuario). La última línea da un número diferente porque PRINT ("PRINP") está mal escrito.

● Los espacios sólo se tienen en cuenta si van entre comillas. Como los espacios entre comillas afectan al programa, también afectan el valor de la suma de control. Los espacios que no vayan entre comillas hacen que el programa sea más legible y, por lo general, se pueden eliminar si quieres. No causan efecto a la suma de control. He aquí unos ejemplos:

```
10 AS="X Z"
```

```
*102
```

```
10 AS="XZ"
```

```
*8
```

Estas dos líneas dan dos distintos valores porque afectan a la ejecución del programa. Observa cómo las dos líneas que vienen a continuación no son iguales, pero su suma de control es la misma. Es porque los espacios no están entre comillas. El efecto es el mismo que si no los pusieras.

```
20 FORX=10TO20
```

```
*30
```

```
20 FOR X = 10 TO 20
```

```
*30
```

No debes incluir espacios entre las palabras Basic reservadas (PRINT, por ejemplo), porque causaría un error en la suma de control. Si tienes algún problema, introduce la línea tal y como aparece en el listado.

● El orden de los caracteres afectará a la suma de control. Las sumas de control que no coincidan previenen la posibilidad de que "bailes" dos letras o números (cosa importante en los pokes). Los siguientes ejemplos aclaran cómo funciona:

```
10 POKE 57,13
```

```
*242
```

```
10 POKE 75,13
```

```
*226
```

# PERFECTOS



Aunque en el ejemplo anterior se utilizan los mismos caracteres, la suma de control es diferente. Lo mismo sucede en el siguiente ejemplo:

```
20 PRINT"STOP"
*106
20 PRINT"POTS"
*94
```

Resumiendo: se pueden abreviar los comandos Basic, los textos (es decir, todo lo que no son comandos Basic) deben coincidir con el listado y los espacios **entre comillas** han de corresponderse exactamente con los del listado.

#### Corrigiendo líneas

Si la suma de control de la línea que acabas de teclear no coincide con la que aparece en el listado, busca algún error que hayas cometido. Esto incluye letras que te hayas comido en textos o en las palabras claves. Compara con cuidado los textos que vayan entre comillas con los del listado.

Si hay espacios entre comillas en el listado, **deben** teclearse cuando se introduzca la línea. Si quieres cambiar el espaciado o incluso el texto, teclea primero la línea para obtener la suma de control correcta y a continuación modifícala. Con esto te asegurarás de que los demás comandos Basic de la línea han sido tecleados correctamente. Cuando tengas ya un programa que funcione podrás añadir o borrar espacios o texto.

Comprueba el número de la línea que estás tecleando porque también influye en el resultado de la suma de control. Si no es correcto, cámbialo. Corrige o borra también las líneas que accidentalmente hayas creado o borrado. Por ejemplo, si estás tecleando la línea 1000 y por error omites un cero (creando de este modo la línea 100) tendrás que volver a introducir la línea 100, y a continuación la 1000.

Si por alguna razón no consigues obtener la suma de control correcta, prueba a borrar la pantalla o a bajar unas cuantas líneas y a teclear la línea completa de nuevo. Puedes haber introducido algún shift-espacio o algo que "parece" que está bien, pero que en realidad está equivocado. Si aún así tampoco consigues obtener la suma de control correcta, apúntala, sigue con el programa y vuelve más tarde a ella.

Después de usar a "Perfecto" la búsqueda de errores en los programas se convertirá en algo del pasado. Ve obteniendo las sumas de control línea por línea y los programas funcionarán a la primera.

```
1 REM "PERFECTO" .113
2 REM POR JAMES E. BORDEN, RUN EEUJ .96
3 REM (C)1986 COMMODORE WORLD .157
4 : .236
5 POKE56,PEEK(56)-1:POKE52,PEEK(56) .119
6 CLR:PG=PEEK(56):ML=PG*256+60 .232
7 : .239
8 P=ML:L=340 .92
9 S=0:FORI=0TO6:READA:IFA=-1THEN16 .59
10 IFA<00RA>255THEN14 .146
11 POKEP+I,A:S=S+A:NEXT .81
12 READSC:IFS<>SCTHEN14 .250
13 L=L+10:P=P+7:GOTO9 .149
14 PRINT"ERROR EN DATAS LINEA";L:END .60
15 : .247
16 POKEML+4,PG:POKEML+10,PG .60
17 POKEML+16,PG:POKEML+20,PG .221
18 POKEML+32,PG:POKEML+38,PG .110
19 POKEML+141,PG .97
20 SYSML:PRINT"[CRSRD] [WHT]CORRECTO .98
R ACTIVADO
21 PRINT SYS"ML"=CONECTAR .5
22 PRINT SYS"ML+30"=DESCONECTAR[COM .122
M7]
23 : .255
24 DATA173,5,3,201,3,208,1,594 .22
25 DATA96,141,105,3,173,4,3,525 .181
26 DATA141,104,3,162,103,160,3,676 .214
27 DATA142,4,3,140,5,3,96,393 .177
28 DATA234,234,173,104,3,141,4,893 .96
29 DATA3,173,105,3,141,5,3,433 .177
30 DATA96,32,124,165,132,11,162,722 .18
31 DATA0,142,240,3,142,241,3,771 .87
32 DATA189,0,2,240,51,201,32,715 .166
33 DATA208,4,164,212,240,40,201,1069 .177
34 DATA34,208,8,72,165,212,73,772 .146
35 DATA1,133,212,104,72,238,241,1001 .237
36 DATA3,173,241,3,41,7,168,636 .142
37 DATA104,24,72,24,104,16,1,345 .225
38 DATA56,42,136,16,246,109,240,845 .238
39 DATA3,141,240,3,232,208,200,1027 .123
40 DATA173,240,3,24,101,20,24,585 .72
41 DATA101,21,141,240,3,169,42,717 .49
42 DATA32,210,255,169,0,174,240,1080 .170
43 DATA3,32,205,189,162,4,189,784 .83
44 DATA211,3,32,210,255,202,16,929 .214
45 DATA247,164,11,96,145,13,32,708 .87
46 DATA32,0,0,0,0,0,0,32,-1 .146
```

# LISTADOS PERFECTOS





# SEINFO, S.L.

SERVICIOS DE INFORMATICA

## PROGRAMAS PROFESIONALES

# commodore 64 128



25.000

### PAQUETE INTEGRADO DE FACTURACION Y CONTROL DE STOCKS

- Capacidades de ficheros programables por el usuario (clientes, artículos, proveedores).
- Control de entradas/salidas de almacén.
- Inventario permanente e inventario bajo mínimos.
- Gestión de reserva de pedidos.
- Facturación y emisión de recibos.
- Posibilidad de facturar artículos no existentes (facturación directa).
- Programación impuesto IVA.
- Listado de entradas/salidas de almacén.
- Listado de pedidos pendientes, diario de ventas, remesa bancaria.
- Listado de ficheros con cabecera programable.
- Listados con criterios de selección de fichas.
- Emisión de etiquetas.
- Conexión con tratamiento de textos (documentación personalizada).

### ESTRUCTURAS

Calcula pórticos planos de hormigón armado. Calcula los esfuerzos para las tres hipótesis verticales, viento y sismo. Armado total de vigas y pilares. Cuadro de pesos de hierro. Cuadro cúbico de hormigón. Listado de todos los esfuerzos en el armado.

25.000

### MEDICIONES Y PRESUPUESTOS

Programa de mediciones y presupuestos de obra totalmente programable por el usuario. Listado de mediciones y presupuesto por partidas. Posibilidad de ajuste automático de presupuesto.

25.000

### CONTABILIDAD

Basada en el Plan Contable Español, 300 ó 1.000 cuentas. Contrapartida automática. Extractos por pantalla o impresora. Balances programables. Grupos 0 y 9. Balance de situación y cuenta de explotación programables.

25.000

### FACTURACION

Programa de facturación directa. Fichero de artículos y clientes. Diarios de ventas. Desglose de impuestos. Emisión de recibos. Varias versiones.

15.000

### CONTROL DE STOCKS

Ficheros de artículos y proveedores. Control de entrada-salida de almacén. Actualización automática. Inventario permanente. Inventario bajo mínimos. Listados varios.

15.000

### CALCULOS ELECTRICOS (II)

Cálculo de redes de alta tensión. Cálculos mecánicos y eléctricos. Cálculo de redes en baja tensión. Informes completos para adjuntar el proyecto.

15.000

### OTROS PROGRAMAS

- |   |              |                                  |              |
|---|--------------|----------------------------------|--------------|
| • BASE DE DATOS (SUPERBASE 64) .....          | 22.500 ptas. | • QUINIELAS .....                | 3.000 ptas.  |
| • TRATAMIENTO DE TEXTOS .....                 | 10.000 ptas. | • ADMINISTRACION DE FINCAS ..... | 35.000 ptas. |
| • ESTADISTICA I, II, III .....                | 3.000 ptas.  | • RECIBOS DE COLEGIOS .....      | 25.000 ptas. |
| • CONTABILIDAD ESPECIAL .....                 | 5.000 ptas.  | • FACTURACION POR ALBARAN .....  | 15.000 ptas. |
| • MODULO DE PASO (Gestión-Contabilidad) ..... | 7.000 ptas.  | • EMISION DE ETIQUETAS .....     | 5.000 ptas.  |

### OTROS:

- CONFECCION DE PROGRAMAS A MEDIDA.
- DESARROLLOS DE HARDWARE.
- DISQUETTES, MONITORES, CINTAS, etc.

Pida información: (976) 22 69 74  
23 29 61

**SEINFO, S.L.**  
SERVICIOS DE INFORMATICA

Avda. Goya, 8 - 50006 ZARAGOZA  
(Distribuidores exclusivamente por carta)



# RESET: inicialización

Por Antonio Gámez

Ahora tenéis la oportunidad de conocer en profundidad todo lo que sucede durante los tres segundos y medio que tarda vuestro ordenador en comenzar a funcionar. Aunque está basado en la ROM del C-64 los demás modelos Commodore funcionan de una forma muy parecida.

**A** la hora de hablar de inicialización de un sistema, el usuario de tipo medio piensa en el desenchufado de la máquina y nueva conexión, cuando esto no es totalmente necesario para provocar una inicialización o RESET a la máquina. Al leer esto los usuarios un poco más avezados en palabras técnicas habrán caído en que aquí nos referimos a esa patilla del port de usuario, port serie, cartucho, que se señala como RESET y que al unirla y volver a separarla con otra patilla denominada GND se produce el mismo efecto que al volver a conectar el C-64, aunque existen programas que no le hacen el más mínimo caso a esta operación. Pero entremos en más profundidades y analicemos qué significa un reset para un sistema y el porqué de extraños comportamientos después del reset en ciertos programas que se resisten a abandonar la memoria de nuestro C-64. Todos estos misterios y muchos otros piensan ser desvelados, aunque no totalmente (nunca se sabe si se está infringiendo algún copyright). Vayamos por partes y empecemos por desvelar el misterio de la operación de reset. Esta operación consiste en un buen número de inicializaciones menores que sirven para poner en funcionamiento al unísono y ordenadamente a todos y cada uno de los circuitos integrantes de C-64, por lo que se deduce que es distinto para cada máquina en particular y afinando un poco más digamos que es diferente en cuanto a cada máquina como sistema y a cada microprocesador como unidad central de proceso que rige y ordena la totalidad del sistema.

Y surge la duda. ¿Qué hace un determinado microprocesador cuando se le conecta y cómo saberlo de una manera fiable?

La respuesta a este enigma que a más de uno se le presentó en su día viene de la mano del máximo responsable del buen funcionamiento de un microprocesador, como es el fabricante. Si, pero ¿cómo encuentro yo al fabricante de un 6510? (ardua tarea). Pues es bien fácil, puesto que en el caso del 6510 sus dos primeras cifras indican la familia a la que

pertenece. Esto, la familia 65XX, cuyos fabricantes son Rockwell y MOS Technology y en sus catálogos de características encontraremos la descripción de las primeras operaciones que realiza un 6510 tras proporcionarle un buen RESET. Para no incordiar mucho a los impacientes hagamos referencias a una publicación española del ilustre José María Angulo Usategui que en su libro "Microprocesadores Diseño Práctico de Sistemas" de Ed. Paraninfo nos ahorra el trabajo de traducir catálogos en lengua no cristiana, siendo además un libro muy recomendable para los que desean iniciarse en estas técnicas. En este libro se nos dice que tras darle un reset el 6502 (el pariente más cercano del 6510) éste dispone de un vector en las posiciones \$FFFC y \$FFFD (el hexadecimal es inevitable en estas cuestiones de alta tecnología). Y un vector no es ni más ni menos que un par de direcciones en las que se encuentran parte baja y parte alta respectivamente de la dirección a la que el microprocesador saltará inapelablemente y de la que recogerá el primer byte tenido en cuenta como operación a realizar por el 6510 en el C-64.

El contenido de estas posiciones de memoria en el C-64 es \$E2 y \$FC respectivamente, con lo cual la dirección formada es \$FCE2 (en decimal 64738), siendo ésta donde se inicia el correr del programa de inicialización.

Hay que apostillar ahora que tanto la rutina de inicialización que está a partir de \$FCE2 como el vector reset deben estar en ROM. Esto lo corrobora el hecho de que el KERNAL esté situado desde E000-FFFF; si el kernal no estuviese en estas direcciones no sería posible la inicialización.

Ya hemos desvelado el primer misterio de una inicialización y el porqué el kernal debe estar en \$E000-\$FFFF. Estas deducciones son compatibles con toda máquina que lleve un microprocesador de esta familia, no con los de otras familias como el Z-80 o el 8088, cuya inicialización es totalmente distinta.

Veamos el siguiente paso una vez sabido todo lo que ocurre a partir de \$FCE2, en donde podremos empezar a desensamblar para ver las operaciones realizadas. El desensamblado se va a realizar para ver las rutinas de inicialización hasta tocar por primera vez la ROM de Basic que está en A000. También hay que señalar antes de empezar que se comentarán algunas rutinas de interés y no todas.

```

..FCE2 LDX #$FF
..FCE4 SEI          ;inhabilita interrupciones IRQ
..FCE5 TXS          ;coloca el stack pointer a $FF
..FCE6 CLD          ;borra el modo decimal
..FCE7 JMP $F905    ;test de cartucho
..FCEA BNE $FCEF    ;no, sigue inicializacion
..FCEC JMP ($8000)  ;si, salta vector en $8000
..FCE2 STX $D016    ;rellena registro control del VIC
..FCF2 JSR $FDA3    ;configuración de CIAs, PAL, interrupciones
..FCF5 JSR $FD50    ;configuración pagina de memoria, test RAM
..FCF8 JSR $FD15    ;configuración de vectores E/S
..FCFB JSR $FF5B    ;rutinas de video
..FCFE CLI          ;habilita interrupciones
..FCFF JMP ($A000)  ;entra al Basic por el vector $A000
    
```



a



**paso**

**C**uando lo conectamos, el C-64 busca en su ROM la dirección de memoria en la que comienzan las rutinas de inicialización.

Veamos algunas curiosidades como son, por ejemplo, que nada más empezar la rutina de inicialización se inhabilitan las interrupciones. Esto es debido a que el microprocesador chequea la entrada de interrupción. Si ésta tuviese lugar, se produciría un salto en el programa, lo que pillaría a la máquina sin inicializar, provocando un cuelgue irremisible del sistema.

Una vez inhabilitadas las interrupciones con el propósito explicado se procede al dimensionado del stack y, por tanto, al llenado del stack pointer con el valor \$FF, con lo que el área de stack queda designado de \$FF hacia abajo, ocupándose este área al producirse saltos de tipo subrutina (pero es innecesario explicar esto aquí cuando ya se ha hecho en el cursillo de lenguaje máquina). Veamos ahora el primer salto a subrutina hacia \$FD02 y veamos qué ocurre allí:

```

..FD02 LDX #505      ;van a compararse 5 caracteres
..FD04 LDA $F00F,X    ;carga acumulador con caracteres
..FD07 CMP #8003,X    ;compara con el homologo en 8003
..FD0A BNE $F00F      ;distinto vuelve
..FD0C DEX            ;no, decrementa offset
..FD0D BNE $FD04      ;va por siguiente, no han pasado 5
..FD0F RTS            ;han pasado 5 caracteres, vuelta
; "CBM80"

```

Esta rutina es la encargada de chequear la existencia o no de un cartucho conectado, y si éste debe ejecutarse o no. En realidad lo que hace la rutina es ver si los caracteres de \$8003 en adelante coinciden en contenido con las siglas "CBM80" en ASCII. Esto supone compararlos con los códigos \$C3, \$C2, \$CD, \$38 y \$30. Si esta comparación es totalmente válida se termina en el registro X con un 00. En realidad esta rutina chequea la zona de \$8000 sea ésta ROM o RAM, ya que de la conmutación se encarga otra circuitería, que no viene al caso comentar. Al volver de esta subrutina a la rutina principal pueden existir dos caminos a seguir:

a) En la zona \$8000 existe el código \$C3, \$C2, \$CD, \$38, \$30, con lo que queda probado que existe cartucho, o al menos un programa que debe ejecutarse. En este caso las rutinas de inicialización son detenidas y se transfiere la ejecución al vector contenido en \$8000 y \$8001 como así lo expresa la ejecución de JMP (\$8000).

b) Si no existe este código se sigue la inicialización y se ignora el contenido del área de \$8000.

Se acaba de desvelar otro secreto: para que un programa se inicie después de darle al reset es necesaria la existencia del código "CBM80" y tener el vector de inicio en \$8000. He aquí el secreto de los famosos juegos que pese a darles un reset siguen literalmente agarrados al C-64 y no desaparecen. También hay que decir que si a alguno se le ha ocurrido hacer un programa de este tipo deberá realizar en su programa las rutinas de inicialización ya que el C-64 dejará de hacerlas en su forma habitual al existir el cartucho.

Vayamos ahora a por la siguiente subrutina situada en \$FDA3:

```

..FDR3 LDA #7F      ;
..FDR5 STA $DC0D     ;configura el modo de tratamiento de
..FDR8 STA $DD0D     ;las interrupciones por parte de
                        ;las CIA 1 y 2
..FDRB STA $DC0E     ;
..FDRF LDA #08      ;
..FDB0 STA $DC0E     ;configura relojes internos de
                        ;las CIAs
..FDB3 STA $DD0E     ;
..FDB6 STA $DC0F     ;establece el modo de trabajo de
..FDB9 STA $DD0F     ;CIA1 y CIA2
..FDBC LDX #08       ;
..FDBF STX $DC03     ;establece el port B de ambas CIAs
..FDC1 STX $DD03     ;de entrada por todas las líneas
..FDC4 STX $D418     ;corte de filtros y volumen de sonido
                        ;a cero (evitando los ruidos)
..FDC7 DEX           ;X = $FF
..FDC8 STX $DC02     ;Port A, CIA 1 todo salida
..FDCB LDA #07       ;
..FDCD STA $DD00     ;inicialización contenido port A CIA1
..FDD0 LDA #3F       ;
..FDD2 STA $DD02     ;establece dirección datos port A CIA2

```

El porqué de estas configuraciones es algo bastante largo de contar y no contiene por sí mismo ninguna curiosidad. Sin embargo los dos pasos que vienen a continuación son vitales para el C-64, como es el establecimiento del mapa de memoria de funcionamiento.

Explicemos esta cuestión con un poco más de detenimiento. Sabemos que el 6510 tiene un puesto interno en el propio microprocesador, a través del cual se le indican qué bloques de memoria puede conmutar de RAM a ROM para realizar su trabajo. Esto hace posible el poder trabajar con 64K de RAM auténtica, sin trampa ni cartón, siendo el Commodore 64 el único ordenador del mercado que puede hacerlo. Esto hay que recalcarlo porque hay gente que todavía no se lo ha creído, pero hay que decir también que estos 64K funcionan sin el BASIC y sin el querido KERNAL, con lo que su manejo queda reservado a un pequeño grupo de usuarios avezados en el diseño de sus propias rutinas y auténticos artesanos del noble quehacer de conmutar bloques, trasladar en memoria y manejar las E/S del C-64 con la habilidad de un relojero y el pulso de un cirujano, pues al menor descuido se puede dar todo por perdido en la confección de un programa de este tipo. Pero veamos qué ocurre ahora:

```

SFDD5 LDA #E7

```

```

SFDD7 STA $01

```

En estas dos simples líneas se carga la posición \$01 con el contenido \$E7, significando esto para el procesador que se recurre al mapa de memoria denominado estándar, es decir, Kernal de \$E000 a \$FFF, \$D000 para chip de video y \$A000 para Basic, siendo estas zonas de ROM y el resto RAM compartida o no en acceso por el VIC o el procesador.

Si en este kernal del que estamos desenredando algunos misterios pudiésemos modificar estas dos líneas y en vez de \$E7 pusiésemos \$E6 tendríamos la sensación de un C-64 desgraciado totalmente, pues habríamos desconectado el BASIC por su zona análoga de RAM que probablemente esté llena de basura y, por tanto, se haría imposible el trabajar con este C-64.

El resto de la rutina hasta su vuelta se dedica a la configuración de las dos CIAs y a la selección del banco de trabajo del VIC.



A continuación se detalla el desensamblado del resto de la rutina regocijo de los curiosos:

```

..FD09 LDA #2F      ;
..FD0B STA $00      ;configura la E/S del puerto del 6518
..FD0D LDA $02A6    ;verifica el sistema de video
..FD0E BEQ $FDEC     ;0=NTSC, 1=PAL
..FD2E LDA #25      ;
..FD4A STA $DC04     ;configura byte bajo timer CIA1 para PAL
..FD47 LDA #440     ;byte alto para PAL
..FD49 JMP $FDF3     ;
..FD4C LDA #95      ;byte bajo para NTSC
..FD4E STA $DC04     ;
..FD51 LDA #442     ;
..FD53 STA $DC05     ;
..FD56 JMP $FF6E     ;

```

Prosigamos por el salto para ver el resto en el que controlan las interrupciones, relojes interiores de CIAS y banco de trabajo del VIC.

```

..FF6E LDA #81      ;
..FF70 STA $DC00     ;trata las interrupciones de CIA1
..FF73 LDA $DC0E     ;
..FF76 AND #80      ;
..FF78 ORA #11      ;
..FF7A STA $DC0E     ;configura relojes interiores de CIA1
..FF7D JMP $EE8E     ;
..EE8E LDA $D000     ;
..EE91 ORA #10      ;
..EE93 STA $D000     ;selecciona banco de trabajo del VIC
..EE96 RTS           ;por fin el retorno

```

Una vez de vuelta a la rutina principal, nos topamos de nuevo con un salto a subrutina siendo ésta una de las más interesantes de conocer ya que en ella se encuentra el famoso test de RAM, tras el cual se nos muestra la flamante cifra de 38911 bytes libres que todo el mundo conoce y que no es totalmente verdadera como ahora veremos:

```

..FD50 LDA #00      ;constante numerica 00
..FD52 TAY           ;variable en Y para su uso como offset
..FD53 STA $0002,Y   ;rellena $0002 a $0002+$FF con 00
..FD56 STA $0200,Y   ;rellena $0200 a $0200+$FF con 00
..FD59 STA $0300,Y   ;rellena $0300 a $0300+$FF con 00
..FD5C INY           ;incrementa Y
..FD5D BNE $FD53     ;repite hasta que Y = 00

```

Con este breve pero curioso inicio se consiguen llenar tres zonas de la RAM de gran utilidad, inicializándolas al poner todas y cada una de sus posiciones a cero, lo que se consigue gracias a la facilidad del indexado.

Si alguien no ve claro estos pasos por favor que recurra a algún artículo que describa el funcionamiento de un direccionamiento indexado antes de seguir, o no se va a enterar de nada.

Prosigamos:

```

..FD5F LDX #3C      ;lee puntero en los registros
..FD61 LDY #03      ;X e Y, y a continuación lo
..FD63 STX $B2      ;coloca en el puntero de inicio
..FD65 STY $B3      ;del buffer cassette; queda $033C
..FD67 TAY           ;

```

Hagamos un alto en esta operación para recapacitar sobre el contenido del registro Y ya que después nos va a hacer falta para un extraño indexado a realizar para la siguiente parte de la rutina. TAY significa trasvasar al registro Y el contenido del acumulador. Si volvemos sobre nuestros pasos descubrimos que la última vez que se rellenó el acumulador fue en \$FD50 LDA#\$00 por lo que Y=\$00. Descifremos ahora una de las rutinas más interesantes y que nos descubre dos grandes novedades: por un lado el que no se destruya el contenido de la RAM y por otro el famoso valor de inicio de RAM en \$0800 y el final de la RAM no del todo cierto:

```

..FD68 LDA #03      ;
..FD6A STA $C2       ;
..FD6C INC $C2       ;establece inicio test ($C2) en 04
..FD6E LDA ($C1),Y   ;$C1 = 0, carga ($0400) -> acumulador
..FD70 TAX           ;acumulador -> X, se guarda contenido
..FD71 LDA #55      ;
..FD73 STA ($C1),Y   ;$55 -> ($0400)
..FD75 CMP ($C1),Y   ;se comprueba que existe un $55
..FD77 BNE $FD88     ;no, fallo en RAM: va a $FD88
..FD79 ROL           ;si, se cambia el contenido por $AA
..FD7A STA ($C1),Y   ;se deposita $AA
..FD7C CMP ($C1),Y   ;se comprueba
..FD7E BNE $FD88     ;no, fallo en RAM
..FD80 TXA           ;si, se devuelve valor anterior al test

```

**E**l chequeo de RAM se realiza sin destruir el contenido de la memoria, pudiéndose de este modo recuperar programas Basic o de código máquina cuando se pierde el control sobre el ordenador.

```

..FD81 STA ($C1),Y   ;se restaura valor
..FD83 INY           ;se incrementa offset
..FD84 BNE $FD8E     ;offset < 0, repite
..FD86 BEQ $FD8C     ;offset = 0 incrementa byte alto en $FD6C
..FD88 TYA           ;fallo en RAM, Y -> A, posición fallida
..FD89 TAX           ;dentro de $FF que se queda en X (A -> X)
..FD8A LDY $C2       ;en Y se guarda el byte alto de la
                    ;posición fallida
..FD8C CLC           ;limpia carry
..FD8D JSR $FE20     ;establece puntero fin de RAM
..FD8F LDA #08       ;
..FD90 LDA #08       ;puntero de inicio de RAM: $0200 = $08
..FD92 STA $0282     ;$0281 = $01
..FD95 LDA #04       ;
..FD97 STA $0298     ;establece número de página final de
                    ;memoria de pantalla
..FD9A RTS           ;vuelta a la rutina principal

```

Subrutina en FE20:

```

..FE20 STX $02B3     ;byte bajo de posición de RAM fallida
..FE22 STY $02B4     ;byte alto de posición de RAM fallida
..FE24 RTS           ;vuelta a la rutina principal

```

El proceso consiste en comprobar la RAM a partir de \$0400 hasta que un byte falle. Para ello se utiliza el indexado indirecto con puntero en \$C1, \$C2; la comprobación se realiza depositando primero un \$55 y ver qué se encuentra allí, después hacer lo mismo con \$AA y si no falla es que la posición de RAM está en buen estado. De lo contrario habremos topado con una posición de ROM o una de RAM defectuosa lo que obligará a la detención del proceso, estableciéndose punteros de inicio y final de RAM, que es la RAM que se puede utilizar por el BASIC, no la totalidad de la RAM existente.

Los más incrédulos pensarán que también será capricho de la calenturienta mente del diseñador el escoger el \$55 y el \$AA para hacer un test de RAM en vez de otros valores. Esto no es realmente así y hay que aclarar el porqué. Lo cierto es que el C-64, al igual que otros sistemas similares, dispone de memorias RAM de tipo dinámico cuya estructura interna en el proceso de almacenaje de información es similar al de la carga de un condensador, por lo que para probar estas estructuras lo recomendable es cargar primero los condensadores y luego descargarlos para ver si la transferencia de carga ocurre. Para ello se carga la RAM con un \$55 que si se ve en binario es 01010101. Pasando después con un simple ROL al valor \$AA, en binario 10101010, se prueba el resto de la carga. Este es el método empleado universalmente para probar las RAM de este tipo.

Es curioso cómo nada más empezar esta rutina se guarda el contenido de su posición a probar en el registro X, volviendo a ser restaurado su valor cuando el test ha tenido un final satisfactorio. Esto tiene un carácter no destructivo, quedando todo como estaba después del test. Se acaba de descifrar el misterio de que no se destruyan los programas contenidos en RAM tras un reset (siendo este punto la envidia de los poseedores de ZX-Spectrum+ o no+). Tras encontrar el fatídico byte fallido se establecen los punteros de RAM. El de inicio se establece en \$0800 y el final en el primer fallo. Esto se producirá en la ROM del BASIC, es decir en \$A000, siendo esta posición de ROM por lo que la RAM queda en \$9FFF, si hacemos la cuenta \$9FFF - \$0800 = \$97FF. Si lo pasamos a decimal nos daría el conocido 38911 que todo el mundo ve al conectar el



aparato. Pero esto no es del todo cierto ya que después del BASIC, en \$C000, queda una zona muerta no utilizable por programas BASIC, pero sí accesible por POKes, para guardar variables o cualquier otra cosa, sabiendo que no van a ser tocadas por el programa BASIC. Si anteriormente se comentó que la RAM quedaba intacta en cuanto a contenido, también hay que decir que su contenido (los programas Basic que estaban en la memoria) no serán listables, pero este fenómeno es independiente del hecho de que un programa en BASIC o en CM seguirá estando en RAM tras un reset y sólo es cuestión de habilidad el conseguir su listado o su volcado sobre disco o cinta, con trucos como el comando OLD.

Tras este comentario, más de uno estará pensando en instalarse un botón de reset con fines no muy licitos. A estos usuarios hay que aclarar que no todos los programas van a ser copiados por este sistema ya que sólo se dejarían cazar aquellos que tras un reset devuelvan el cursor y el control al usuario, aunque hay otros programas que además de no devolver el cursor proseguirán su funcionamiento por medios ya comentados anteriormente.

El botón de reset se consigue conectando las patillas 6 y 2 del port serie del ordenador o también las patillas 1 y 3 del port del usuario. En el manual del usuario aparece la ubicación de ambos contactos. El port de expansión también tiene un reset, entre las patillas A y C.

Pero volvamos a la rutina principal para ver que nos quedan un par de subrutinas.

Vayamos por la penúltima en \$FD15 también de un interés relevante:

```

..FD15 LDX #30      ;
..FD17 LDY #FD      ;
..FD19 CLC          ;limpia carry
..FD1A STX $C3      ;configura Tem LOAD
..FD1C STY $C4      ;configura Tem LOAD
..FD1E LDY #1F      ;offset en IV
..FD20 LDA $0314,Y ;
..FD23 BCS $FD27    ;
..FD25 LDA ($C3),Y ;carga de $FD4F a FD30
..FD27 STA ($C3),Y ;
..FD29 STA $0314,Y ;deposita vectores
..FD2C DEY          ;
..FD2D BPL $FD20    ;
..FD2F RTS          ;vuelta a la rutina principal

```

Esta rutina deposita en su lugar correspondiente de RAM; esto es desde \$0314 en adelante, los siguientes vectores:

- Vector interrupción IRQ CINV usado para programar con interrupciones.
- Vector interrupción por BRK CBINV.
- Vector NMI NMIV.
- Vector de la rutina OPEN IOPEN.
- Vector de la rutina CLOSE ICLOSE.
- Vector de la rutina CHKIN ICHKIN.
- Vector de la rutina CHKOUT ICHKOUT.
- Vector de la rutina CLRCHN ICLRCH.
- Vector de la rutina CHRIN IBASIN.
- Vector de la rutina CHROUT IBSOUT.

Así pues estas son rutinas que pueden ser usadas por el usuario mediante llamada, sabiendo claro donde están las condiciones de entrada en la rutina y la salida de la misma.

**L**os punteros de inicio y final del Basic se ajustan después de realizarse el chequeo. De los 64K de memoria RAM que tiene el ordenador sólo 38K pueden ser utilizados por el Basic.

Todo esto puede ser conocido en el libro de referencia del programador que edita Microelectrónica y Control y que detalla con todo rigor el modo de aprovecharlas, así como su funcionamiento.

Vamos por fin de vuelta a la rutina principal en la cual nos encontramos con la última llamada a subrutina de la cual se detalla a continuación su desensamblado haciendo comentario en aquellas partes de algún interés especial y, cómo no, dejando el resto para la práctica del lector. No se hace un comentario completo ya que en esta rutina entran operaciones de pantalla que son largas de explicar, no siendo especialmente necesario su conocimiento para la comprensión de la inicialización de un C-64.

```

..FF5B JSR $E518
..FF5E LDA $D012
..FF61 BNE $FF5E
..FF63 LDA $D019
..FF66 AND #301
..FF68 STA $02A6
..FF6B JMP $FDD0

```

```

..FDD0 LDA $02A6
..FDE0 BEQ $FDEC
..FDE2 LDA #25
..FDE4 STA $DC04
..FDE7 LDA #40
..FDE9 JMP $FDF3
..FDEC LDA #95
..FDEE STA $DC04
..FDF1 LDA #42
..FDF3 STA $DC05
..FDF6 JMP $FF6E

```

```

..FF6E LDA #81
..FF70 STA $DC00
..FF73 LDA $DC0E
..FF76 AND #80
..FF78 ORA #11
..FF7A STA $DC0E
..FF7D JMP $EE8E

```

```

..EE8E LDA $D000
..EE91 ORA #10
..EE93 STA $D000
..EE96 RTS

```

Salto a \$E518:

```

..E518 JSR $E5A0
..E51B LDA #300
..E51D STA $0291 ;desactivar teclas SHIFT
..E520 STA $CF    ;cursor apagado
..E522 LDA #48
..E524 STA $028F ;activacion de la tabla
..E527 LDA #EB    ;del teclado
..E529 STA $0290 ;
..E52C LDA #0A    ;
..E52E STA $0289 ;tamano buffer del teclado = 10
..E531 STA $028C ;contador de retraso de repeticion
..E534 LDA #0E    ;
..E536 STA $0286 ;color actual = azul luminoso
..E539 LDA #04    ;
..E53B STA $028B ;contador de velocidad de repeticion
..E53E LDA #0C    ;
..E540 STA $CD    ;parpadeo entre cursor
..E542 STA $CC    ;parpadeo de cursor
..E544 LDA $0288
..E547 ORA #80
..E549 TAY
..E54A LDA #80
..E54C TAX
..E54D STY $09,X
..E54F CLC
..E550 ADC #28
..E552 BCC $E555
..E554 INY
..E555 INX
..E556 CPX #1A
..E558 BNE $E54D
..E55A LDA #FF
..E55C STA $09,X
..E55E LDX #18
..E560 JSR $E9FF
..E563 DEX
..E564 BPL $E560
..E566 LDY #80
..E568 STY $D3
..E56A STY $D6
..E56C LDX $D6
..E56E LDA $D3
..E570 LDV $D9,X
..E572 BMI $E57C
..E574 CLC
..E575 ADC #28
..E577 STA $D3
..E579 DEX
..E57A BPL $E570
..E57C JSR $E9F0
..E57F LDA #27

```



**E**l botón Reset que algunos ordenadores llevan ya incorporado se puede instalar fácilmente en un C-64.

```

.E581 INX
.E582 LDY #09,X
.E584 BMI $E58C
.E586 CLC
.E587 ADC #528
.E589 INX
.E58A BPL $E582
.E58C STA $D5
.E58E JMP $EA24

```

```

.EA24 LDA #D1
.EA26 STA #F3
.EA28 LDA #D2
.EA2A AND #03
.EA2C ORA #08
.EA2E STA #F4
.EA30 RTS

```

Salto a \$E9F0:

```

.E9F0 LDA $ECF0,X
.E9F3 STA #D1
.E9F5 LDA #09,X
.E9F7 AND #03
.E9F9 ORA #0288
.E9FC STA #D2
.E9FE RTS
.E9FF LDY #527
.EA01 JSR $E9F0
.EA04 JSR $EA24
.EA07 JSR $E4DA
.EA0A LDA #20
.EA0C STA (<D1),Y
.EA0E DEY
.EA0F BPL $EA07
.EA11 RTS

```

Salto a \$E5A0:

```

.E5A0 LDA #03
.E5A2 STA $9A ;configura periférico normal de salida
.E5A4 LDA #00
.E5A6 STA $99 ;configura periférico normal de entrada
.E5A8 LDX #2F
.E5AA LDA $ECB8,X
.E5AD STA $CFFF,X
.E5B0 DEX
.E5B1 BNE $E5AA
.E5B3 RTS

```

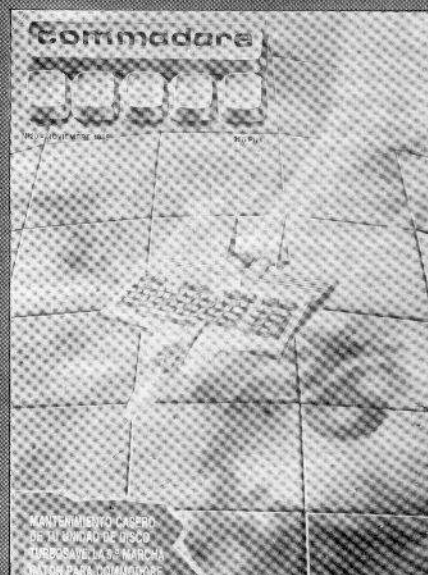
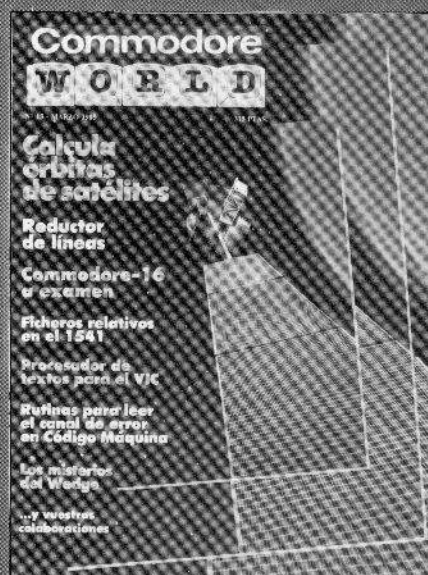
Después de esta ristra de ensamblador se vuelve a la rutina principal en donde encontramos la habilitación de nuevo de las interrupciones y el salto indirecto hacia la ROM del BASIC.

Como últimos comentarios a esta breve introducción al Kernal, decir que todos los cálculos así como el desensamblado han sido realizados ¡a mano! por lo que seguramente contendrá algún error\* que sabrán disculpar. Y para los curiosos diré que todo lo comentado apenas si supone unas décimas de segundo en tiempo de ejecución y que tan sólo hemos cubierto el área de \$E518 a \$FFFF. Tampoco lo hemos estudiado en su totalidad, pues quedan algunas lagunas, y del KERNAL aún queda desde \$E000 a \$E518, que se queda en el tintero para comentarios en otra ocasión.

Espero no haber sido excesivamente profundo y sí haber contribuido a despejar algunas dudas sobre el tema.

*\* Efectivamente, los listados originales estaban escritos a mano y había algún que otro error. Nosotros los hemos sacado directamente de la ROM del C-64 y ya no tienen fallo alguno.*

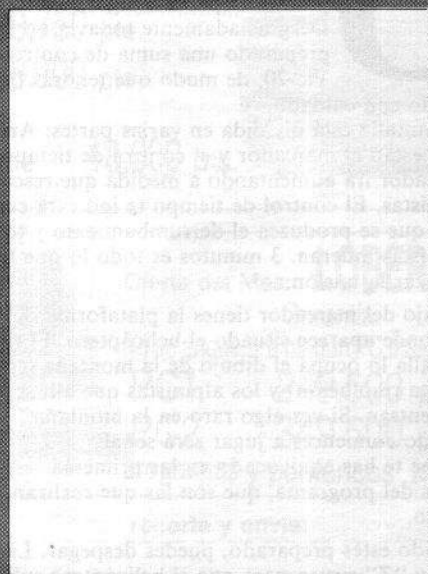
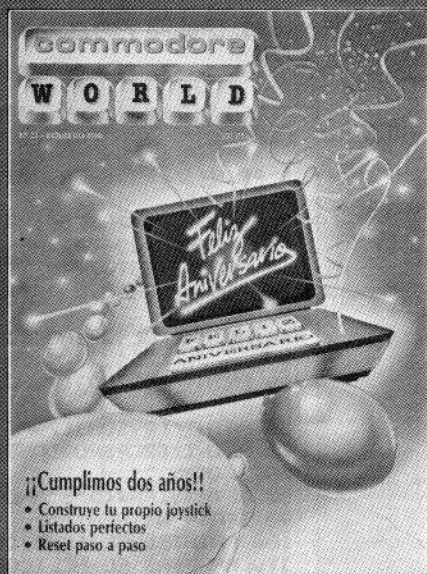
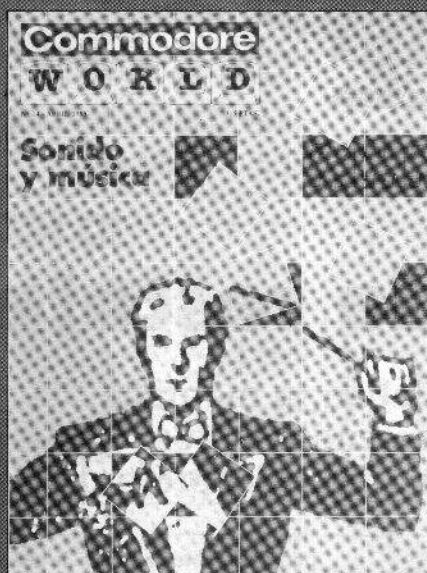
# Si hay hueco en tu biblioteca Commodore





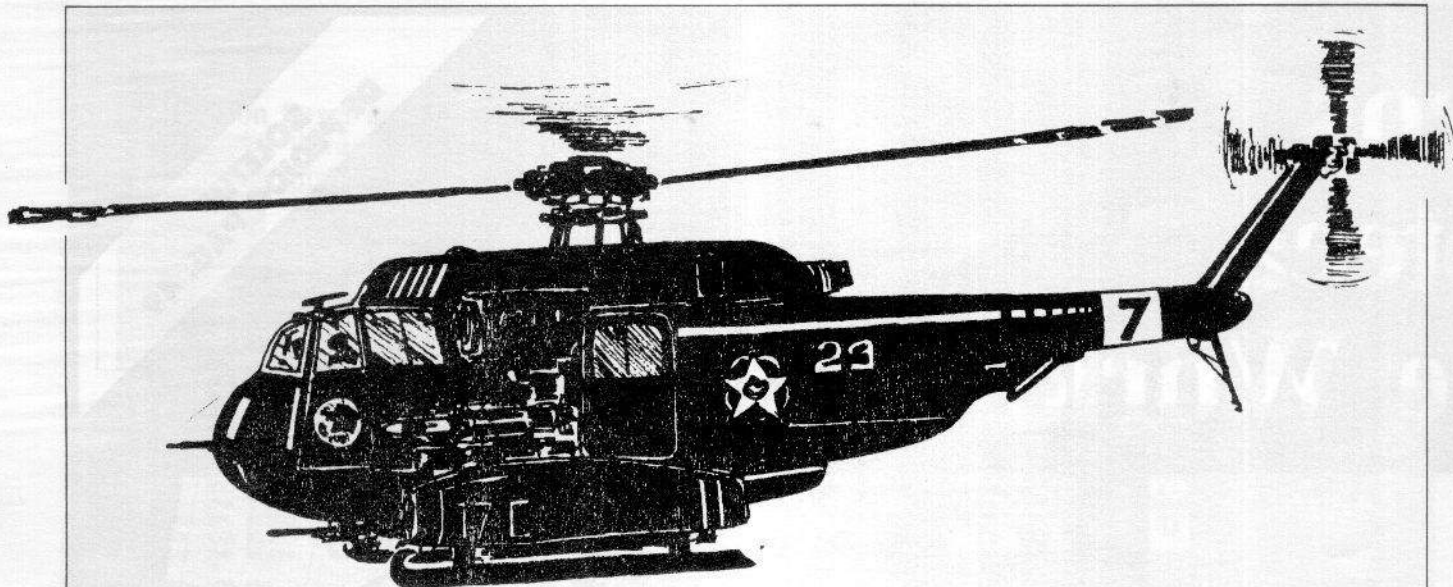
# ecos oteca re World...

BOLETIN  
DE PEDIDO PAG. 49



...rellénalos!!!





# Helicóptero de rescate

Unos alpinistas, mientras escalaban la montaña, han quedado atrapados a causa de unos desprendimientos. La única manera de rescatarlos es por vía aérea, con tu helicóptero de rescate.

**D**espués de que hayas tecleado los dos listados que aparecen al final del artículo podrás empezar a jugar. Desgraciadamente todavía no hemos preparado una suma de control para Vic-20, de modo que tendrás que teclear el

listado con cuidado.

La pantalla está dividida en varias partes: Arriba del todo están el marcador y el control de tiempo. El marcador irá aumentando a medida que rescatas a los alpinistas. El control de tiempo te indicará cuánto queda para que se produzca el derrumbamiento y todos los alpinistas mueran. 3 minutos es todo lo que tienes para finalizar la misión.

Debajo del marcador tienes la plataforma de aterrizaje. Es donde aparece situado el helicóptero. El resto de la pantalla lo ocupa el dibujo de la montaña (con gráficos en alta resolución) y los alpinistas que allí se encuentran. Si ves algo raro en la montaña cuando comiences a jugar será señal de que te has equivocado en las primeras líneas del programa, que son las que realizan el dibujo.

Cuando estés preparado, puedes despegar. Las teclas "A" y "Z" sirven para que el helicóptero suba y baje y las teclas "<" y ">" sirven para moverse a izquierda y derecha. El helicóptero puede moverse hacia atrás, es decir que si se está moviendo hacia la derecha puede

perfectamente cambiar de sentido e ir hacia la izquierda. Para rescatar a los hombrecillos sólo hay que situarse sobre ellos, teniendo cuidado para no chocarse con las paredes. ¡Animo y a rescatarles en menos de tres minutos!

## Listado 1

```
5 PRINT"[CLR][7CRSRD][CRSRR]ESPERA POR F
AVOR..."
7 POKE52,28:POKE56,28:FORA=7168T07679:PO
KEA,PEEK(A+25600):NEXT
10 FORA=7168T07248
20 READB:POKEA,B
30 NEXT
40 PRINT"[CLR]":POKE36879,8:POKE36869,25
5
50 PRINT"[CRSRD][RVSON][WHT][7SPC]CHOPPE
R"
60 PRINT"[RVSON][7SPC][7COMMT]"
70 PRINT"[2CRSRD][CRSRR][RVSON]INSTRUCCI
ONES (S/N)?"
80 GETA$:IFA$<"S"AND A$<"N"THEN80
90 IFA$="N"THEN155
100 PRINT"[YEL][CLR][CRSRD][CRSRR][RVSON
]TU MANEJAS EL HELICOPTERO: [RVSOFF][WHT
]@A[RVSON][YEL]"
110 PRINT"[CRSRD][CRSRR][RVSON][GRN]TIEN
ES QUE RESCATAR A LOS ALPINISTAS PERDIDO
S <[RVSOFF][WHT]G[RVSON][GRN]".
120 PRINT"[RVSON][2CRSRD][CRSRR][CYN]A=[
PUR]ARRIBA[8CRSRL][CRSRD]Z=[CYN]ABAJO[7C
RSRL][CRSRD]<=[GRN]I[QUIERDA[11CRSRL][CR
SRD]>=[YEL]DERECHA"
```

VIC-20 sin expansión



(952) 27 30 43  
39 44 08**commodore**  
COMPUTER

Pts.

C-128 K 69.999  
a sólo

Unid. disco 15% ... 59.990

128/1570 ..... 125.500

IMPRESORA MARGARITA CBM 1101 79.990  
MONITOR color CBM 1901 75.990**SINCLAIR QL**

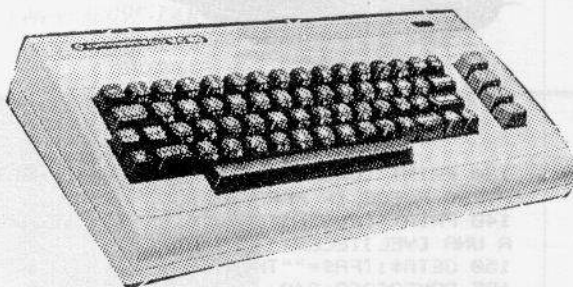
128 K - 32 Bit - 4 programas

59.<sup>900</sup>  
pts.

un precio increíble

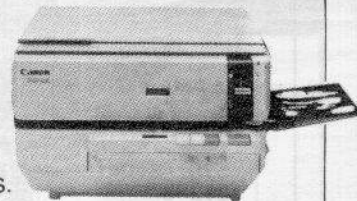
**COMPRUEBA NUESTROS  
SUPER PRECIOS**

12% IVA ya incluido

**COMMODORE 64/1541**  
84.990 PTS.C64 ..... 42.990 pts.  
1541 ..... 42.990 pts.  
Datassette 1531 ..... 8.990 pts.  
Autocopy ..... 3.990 pts.  
Joystick JY 101 ..... 2.500 pts.**Canon****CANON**

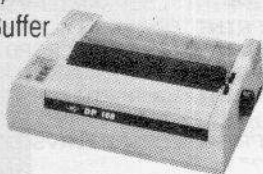
PC 20 189.750 pts.

PC 25 208.900 pts.

Cartucho PC  
negro, rojo, verde 18.600 pts.**IMPRESORA DP165**80 Col. 165/200 CPC, paral/ser  
Interf., tracción-fricción, IBM/  
EPSON Compatible, 2/8 K Buffer

109.950 pts.

Interf. ser. 9.990 pts.

**BASF**

5 1/4" 1 cara doble dens. 350\* pts.

2 cara doble dens. 425\* pts.

3" Maxell 999 pts.

\* min. 1 caja de 20 diskettes.

GARANTIA: 6 meses

ENTREGA: 40 horas

MANUAL usuario:

Castellano

¡COMERCI

CONSULTENOS!



Oferta del Mes:

**LOBERSA PC**

Garantía:

No se encuentra ningún PC más económico

PC1 249.500 pts.

PC2 349.500 pts.

programas y periféricos ¡consultar!

recorte y envíe:

Nombre .....  
Apellido .....  
Dirección .....  
Población .....  
D.P. .... Telef. ....LOBERCIO, S.A.  
COMPOSIT. LHEMB. RUIZ 1  
29007 MALAGA

deseo recibir el material indicado.

pedidos por teléfono

fecha:

firma:

(952) { 27 30 43  
39 44 08  
44 82 64

Cantidad	Artículo	Pts.	total

- ☐ adjunto talón nominativo  
☐ pagaré contra reembolso

Pts.  
totales





```

130 PRINT"[CRSRD][CRSRR][RVSON]EL HELICO
PTERO PUEDE IR PARA ATRAS"
140 PRINT"[RVSON][CRSRD][CRSRR][YEL]PULS
A UNA [YEL]TECLA!"
150 GETA$:IFA$="" THEN150
155 POKE36869,240
160 PRINT"[CLR][RVSON][CRSRD][YEL]CARGA
LA SEGUNDA PARTE[RVSON][5SPC][RVSON]DEL
PROGRAMA":PRINT"[5SPC][RVSON]Y HAZ 'RUN
'." :END
60000 DATA15,0,65,99,127,1,0,7
60010 DATA248,128,240,72,120,240,68,248
60020 DATA0,0,8,140,237,255,255,255
60030 DATA255,255,254,252,248,248,240,19
2
60040 DATA252,252,248,240,248,248,252,25
2
60050 DATA255,255,255,255,255,255,255,25
5
60060 DATA128,192,224,224,248,252,254,25
5
60070 DATA88,88,126,126,26,26,60,102,195
60080 DATA255,255,255,239,237,204,136,0,
0
60090 DATA255,127,7,31,127,7,127,15

```

#### Listado 2

```

1 REM <C> J.G.J.
10 POKE36878,15:POKE36879,8:POKE36869,25
5:HI=0:Q=100:A=3:B=1:CO=38400:POKE650,12
8:Z=0
15 GOT0390
20 PRINT"[RVSON][CLR][9CRSRR]TID:";RIGHT
$(TI$,3)
30 PRINT"[CRSRD][YEL]BBBB[RVSON][RED][3CO
MMI][RVSON][YEL]BBBB[7SPC]BBBBBBB";
40 PRINT"EEEEEEEEEC[7SPC]IEEEEE";
50 PRINT"EEHHHHHH[8SPC]HHHHHEE";
60 PRINT"EED[3SPC][CYN]G[YEL][13SPC]IE";
70 PRINT"EED[2SPC]IED[8SPC]ID[2SPC]IE";
80 PRINT"EED [CYN]G[YEL] ID[8SPC]ID [CYN
]G[YEL]IE";
90 PRINT"EEEEEEEEED[8SPC]IEEEEE";
100 PRINT"CHHHHHHH[8SPC]HHHHHEE";
110 PRINT"F[19SPC]IE";
120 PRINT"D[2SPC]IBBB[2SPC]IBBB[2SPC]ID
[2SPC]IE";
130 PRINT"D[2SPC]IEEC[3SPC]IEED[2SPC]ID[
2SPC]IE";
140 PRINT"D[2SPC]IEEF[3SPC]IEED[2SPC]ID[
2SPC]IE";
150 PRINT"D[2SPC]IEED[3SPC]IEEC[2SPC]ID[
3SPC]IE";
160 PRINT"D[2SPC]HHED[4SPC]ID[3SPC]ID[2S
PC]IE";
170 PRINT"D[3SPC][CYN]G[YEL]IEF[3SPC]ID[
2SPC]BED[2SPC]IE";

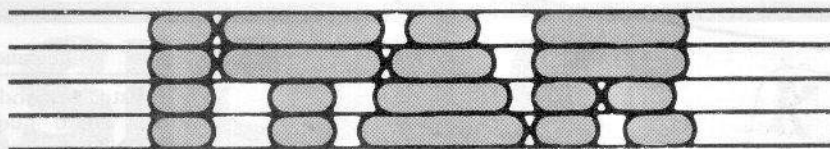
```

```

180 PRINT"D[2SPC]BBBBEEF[2SPC]ID[2SPC]IED
[2SPC]IE";
190 PRINT"D[2SPC]IEEEEC[2SPC]ID[2SPC]IED
[2SPC]IE";
200 PRINT"D[2SPC]IED[CYN]G[YEL][4SPC]ID[
2SPC]IED[2SPC]IE";
210 PRINT"D[2SPC]IEEBBBB[2SPC]IED [CYN
]G[YEL]IE";
220 PRINT"D [CYN]G[YEL]IEEEEEEEEEED [CYN]G
[YEL]IEEBBEE";
230 PRINT"EBEEEEEEEEEEEEEBEEEEEE";
240 PRINT"[HOM][RVSON][PUR]SC="
245 POKE7680+A+22*B,0:POKE7680+A+23*B,1
250 POKE8164+21,5:POKE38905,7
255 POKE36876,130:POKE36875,130
260 GETA$:IFA$="" THEN340
262 POKE7680+A+22*B,32:POKE7680+(A+1)+22
*B,32:POKECO+A+22*B,1:POKECO+(A+1)+22*B,
1
270 IFA$="A" THENB=B-1:IFB<2 THENB=2
280 IFA$="Z" THENB=B+1:IFB>20 THENB=21
290 IFA$="," THENA=A-1:IFA<1 THENA=1
300 IFA$="." THENA=A+1:IFA>19 THENA=19
310 IFPEEK(7680+A+22*B)<>32 ANDPEEK(7680+
A+22*B)<>7 THEN370
320 IFPEEK(7680+(A+1)+22*B)<>32 ANDPEEK(7
680+(A+1)+22*B)<>7 THEN370
321 IFPEEK(7680+A+22*B)=7 THENGOSUB500
322 IFPEEK(7680+(A+1)+22*B)=7 THENGOSUB50
0
330 POKE7680+A+22*B,0:POKE7680+(A+1)+22*
B,1:POKECO+A+22*B,1:POKECO+(A+1)+22*B,1
340 FORW=1 TOQ:NEXT
350 POKE36876,0:POKE36875,0
351 PRINT"[WHT][RVSON][HOM][9CRSRR]TID:"
;MID$(TI$,4,1)". "MID$(TI$,5,4)
352 IFTI$="000300" THENGOTO2000
360 GOTO255
370 POKE36876,0:POKE36875,0
371 FORB=8 TO255 STEP8:POKE36879,B:FORA=1 T
O15:POKE36878,A:POKE36877,B:NEXTA,B:POKE
36877,0
380 IFSC>HI THENHI=SC
390 PRINT"[CLR]":POKE36879,8
400 PRINT"[CRSRD][RVSON][5CRSRR]*****
***":PRINT"[5CRSRR][RVSON]* CHOPPER *"
410 PRINT"[RVSON][5CRSRR]*****"
420 PRINT"[RVSON][CRSRD][WHT]PREPARE P
ARA MORIR !"
430 PRINT"[3CRSRD][CRSRR][RVSON]HIGH SCO
RE ";HI
440 PRINT"[CRSRD][RVSON]TE ATREVES OTRA
VEZ ?"
460 GETA$:IFA$<"S" ANDA$<"N" THEN460
470 IFA$="S" THENA=3:B=1:Q=100:SC=0:Z=0:T
I$="000000":GOTO20
480 END
500 POKE36878,15
510 FORL=1 TO15
520 FORM=240 TO240 STEP-1
530 POKE36876,M0
540 NEXTM
550 FORM=240 TO250
560 POKE36876,M
570 NEXTM
580 POKE36876,0
590 NEXTL
600 SC=SC+10:PRINT"[HOM][3CRSRR][RVSON]
[WHT]"SC:Z=Z+1:IFZ=8 THENZ=0:A=3:B=1:Q=Q-
5:GOTO20
610 RETURN
2000 POKE36879,8:PRINT"[CLR][RVSON][CRSR
D][5CRSRR][CYN]*****":PRINT"[5CRSR
R][RVSON]* CHOPPER [RVSON]*"
2010 PRINT"[RVSON][5CRSRR]*****"
2020 PRINT"[CRSRD][RVSON][WHT]SE TE ACAB
O EL TIEMPO!":GOTO430

```





COMPUTERS, S.A.

**PAMPLONA:** Alfonso el Batallador, 16 (trasero) - Teléfono: 27 64 04 - Código Postal: 31007

**SAN SEBASTIAN:** Plaza de Bilbao, 1 - Tel. 42 62 37 - Télex: 38095-IART - Cód. Postal 20005



**CONDICIONES ESPECIALES PARA DISTRIBUIDORES**





### Los nuevos comandos

Estos comandos no son imprescindibles como puede ser, por ejemplo, PRINT; no son realmente necesarios pero ayudan de vez en cuando. Las ventanas realzan el aspecto de una pantalla y pueden ser usadas como títulos, para subrayar mensajes de error, y otras cosas parecidas.

Todos los comandos de ventanas requieren que se especifiquen por lo menos cuatro parámetros, aunque algunos necesitan más. Dos de los parámetros son las coordenadas X e Y de la parte superior izquierda de la ventana (X = columna de la izquierda; Y = primera línea). Los otros dos parámetros especifican la anchura y la profundidad de la ventana. Para cuestiones de presentación en pantalla, 0,0 indica la esquina superior izquierda de la pantalla (la posición HOME), y 39,24 significa la esquina inferior derecha. Por lo tanto, 0,24 significa la esquina inferior izquierda.

La anchura significa el número de columnas incluidas en la ventana, y puede variar entre 1, el mínimo, y 40, la pantalla entera. La suma de la coordenada X y la anchura debe ser menor o igual que 40. La profundidad indica el número de línea de "altura" de la ventana. Esto puede variar de 1, el mínimo, a 25, la profundidad máxima de la pantalla. La suma de la coordenada y la profundidad debe ser menor o igual que 25.

### Comandos de ventanas

**FILL** valor del código de pantalla, color, x, y, ancho, alto. Este comando rellena la ventana elegida con un carácter del color especificado. Este comando también es capaz de borrar una ventana rellenándola con el carácter espacio. El valor del código de pantalla puede variar entre 0-255. Para determinar el valor correcto del código de pantalla correspondiente al carácter que desees utilizar y el valor numérico del código de color, consulta la guía del usuario. Los valores para los colores pueden ser de 0-15.

Ejemplos:

**FILL 1,8,0,0,40,12.**—Rellena la parte superior de la pantalla con el carácter "A" en naranja.

**FILL 32,8,20,13,20,12.**—Borra solamente un carácter de la pantalla y rellena la memoria de color con el valor para el naranja. Cualquier carácter introducido mediante un Poke en esa posición se verá de color naranja.

**FLASH** (destellea) número de veces, velocidad del parpadeo, columna, fila



# BASIC 4.5

Tercera y última parte.

Por Robert Rockefeller

Este mes vamos a hablar del resto de los comandos del Basic 4.5, empezando por los comandos de ventanas, llamados así porque permiten una manipulación de porciones de la pantalla, o "ventanas". Estas ventanas pueden hacer un scroll en cualquiera de las cuatro direcciones, pueden ser invertidas, hacerlas destellar, ser borradas o rellenadas con un carácter específico.

superior, ancho, alto.—Este comando hace destellar la ventana el número de veces indicado a una determinada velocidad. La ventana puede destellar hasta 255 veces, con una variación en la velocidad entre 1, el mas lento, hasta 255, el más rápido. FLASH significa invertir y volver a invertir, produciéndose así el destello en las ventanas. (Puedes obtener el mismo efecto en un programa usando los comandos REV e INV alternativamente.)

Ejemplo:

**FLASH 4,20,0,0,40,1.**—Hace destellar la fila superior de la pantalla cuatro veces. Este comando es útil para llamar la atención sobre un mensaje de error o para hacer un aviso.

Los siguientes comandos son complementarios; es decir, pueden ser usados los dos a la vez.

**REV** columna, fila, ancho, alto.—Invierte todos los caracteres dentro de la ventana definida. Todos los caracteres saldrán como si hubieran sido escritos en una cadena después del carácter RVS-ON. REV funciona del siguiente modo: repasa cada byte de pantalla dentro de la ventana y enciende el bit 7 de cada uno. Esta es la señal para que el chip de video escriba el carácter en video inverso.

Una advertencia. Cuando se borra la pantalla en los modelos más recientes del C-64, se llena la memoria de color con el valor del color en pantalla en ese momento, y la memoria de pantalla se llena con espacios. Los espacios, que son del mismo color que la pantalla, no se ven nunca, aunque se inviertan, así que si piensas invertir una ventana entera —incluyendo todos los espacios— tienes que borrar la ventana con el

comando FILL antes de escribir cualquier cosa. Elige el color que quieres para los espacios invertidos al usar el comando FILL. La misma advertencia es aplicable al comando INV, que viene a continuación.

**E**sta es la tercera parte de Basic 4.5, un programa que supera al Basic 4.0 con más comandos adicionales sobre música, ventanas, sprites y comandos de disco. Todos los comandos de Basic 4.5 tienen algo en común: Ahorran al usuario una gran cantidad de POKES. Este es uno de los grandes inconvenientes del Basic 2.0 y del 4.0.

Ejemplos:

**PRINT CHR\$(147) : REV 0,1,40,1.**—Borra la pantalla, y luego invierte solamente la segunda línea de la pantalla. Dará la sensación de que no ha ocurrido nada, ya que los espacios invertidos son



del mismo color que la pantalla.

**FILL 32,8,0,1,40,1 : PRINT CHR\$(19) CHR\$(17) "TEST" : REV 0,1,40,1.**—Borra la segunda línea de la pantalla con el comando FILL. La memoria de color utilizada por la segunda línea de la pantalla se llena con el valor 8, que indica el color naranja. Si el color de la pantalla no es naranja, la segunda línea estará en el campo invertido naranja.

**INV** columna, fila, ancho, alto.—Si la ventana definida no contiene caracteres invertidos, la operación de INV es exactamente igual que REV. Si la ventana ya está invertida, el comando INV la pondrá "al revés". El comando INV funciona activando el bit 7 de cada byte de la pantalla. Si en ese momento el bit está activado, queda desactivado y si está desactivado, se activa. Aunque parece igual que REV, no lo es, ya que REV invierte siempre mientras que INV además de esto coloca lo que ya está invertido en modo no-invertido. (¡vaya lío!).

## Hacer scroll y flash

Los cuatro comandos siguientes son para hacer scroll; operan de la misma manera, excepto que el scroll se realiza en cuatro direcciones diferentes. La dirección podrá deducirse del nombre que lleva. Ver los programas de demostración para los ejemplos de uso.

**RIGHTW** columna, fila, ancho, alto. Hace un scroll con el texto dentro de la ventana un carácter a la derecha. Se inserta una columna de espacios en la columna de la izquierda de la ventana. Debes ejecutar el programa de demostración para ver cómo funciona.

**LEFTW** columna, fila, ancho, alto.—



Hace un scroll con el texto de dentro de la ventana una columna hacia la izquierda. La columna de la derecha se rellena de espacios.

**UPW** columna, fila, ancho, alto.— Hace un scroll con el texto de dentro de la ventana una línea hacia arriba. Se inserta una fila de espacios en la última línea de la ventana.

**DOWNW** columna, fila, ancho, alto.— Hace un scroll con el texto de dentro de la ventana una fila hacia abajo. Se inserta una línea de espacios en la primera línea de la ventana.

#### Comandos de pantalla

Los tres comandos siguientes no operan dentro de ventanas, pero están relacionados con la pantalla.

**BFLASH** número de veces, velocidad del parpadeo, color1, color2.—Hace destellar el borde de la pantalla con los colores especificados. Se salva el color original del borde, y luego los dos colores especificados se almacenan alternativamente en el registro del color del borde en la dirección 53280. Después de haberse producido el destello el número de veces especificado, el color original del borde vuelve al registro del color del borde.

El borde puede destellar hasta 255 veces, y la velocidad puede variar de 1 a 255. Aquí, de nuevo, esto puede servir para llamar la atención sobre un error.

Ejemplo:

**BFLASH 5,20,2,10.**—Hace destellar el borde cinco veces, alternando entre rojo y rosa.

**CENTRE** (expresión de cadena).—centra la cadena en la línea de pantalla elegida por el cursor. Si la longitud de la cadena es mayor de 40 caracteres, la acción es la misma que al usar el **PRINT**. **CENTRE** acepta cualquier expresión de cadena utilizable por **PRINT**. Si más de una expresión de cadena sigue a **CENTRE**, el comando solamente operará en la primera.

Ejemplos:

**CENTRE AS**

**CENTRE LNS (5)**

**CENTRE "UNA CADENA"**

**CENTRE "SEGUNDA";AS;A;B**

**LOCATE x, y.**—Este comando coloca el cursor en el sitio donde tú lo quieras poner. Al igual que los comandos de ventanas, 0,0 representa la parte superior izquierda de la pantalla; 39,24 indica la parte inferior derecha.

Ejemplo:

**LOCATE 5,20**

#### Comandos musicales

A continuación vamos a considerar los seis comandos musicales.

**VTYPE** número de voz, forma de onda (**RING** — en anillo, opcional) (**SYNC** — sincronismo, opcional).— Este comando fija el tipo de voz para

**Las ventanas siempre tienen el mismo formato: x, y, ancho, alto. Se colocan en cualquier lugar de la pantalla y pueden invertirse o hacer que parpadeen. Los comandos disponibles son: FILL, FLASH, REV, INV, RIGHTW, LEFTW, UPW y DOWNW. También hay comandos para manejar la pantalla. BFLASH, CENTRE y LOCATE.**

una de las tres voces, y no se vuelve a modificar hasta que no se ejecuta de nuevo el comando **VTYPE**. Los comandos **NOTE** y **PLAY** requieren que se fije el tipo de voz para poder funcionar correctamente. El número de voz puede ser un número de 1 a 3.

El segundo parámetro especifica la forma de onda que será utilizada. El chip de sonido **SID** del C-64 permite cuatro tipos diferentes de forma de onda: triángulo, diente de sierra, pulsación y ruido, seleccionando el tipo usando las cuatro primeras letras del nombre en inglés: **TRI**, **SAW**, **PUL** o **NOI**. Si se selecciona la forma de onda de pulsación, hay que utilizar un número detrás de **PUL**. Este número puede ser cualquier valor de 0 a 2047. Conviene ver los ejemplos a continuación.

Finalmente, existen dos parámetros opcionales. Si se incluye **RING** en la cadena de parámetros, la voz será colocada en el modo de "modulación en anillo" tal y como sugiere su nombre, resulta útil para crear sonidos de campanas. **SYNC** en la cadena de parámetros coloca la voz en el modo de "sincronización". La "Guía de Referencia para el Programador del C-64" utiliza el modo de sincronización para crear el sonido del mosquito. Si quieres más información sobre las capacidades del chip **SID**, la Guía dedica 24 páginas al tema.

Ejemplos:

**VTYPE 1, PUL1000.**—Elige la forma de onda de pulsación para la voz 1.

**VTYPE 2, TRI, SYNC.**—Elige la forma de onda triangular y el modo de sincronización para la voz 2.

**VTYPE 3, SAW,RING,SYNC.**—Elige

la forma de onda de diente de sierra, el modo de modulación en anillo, y el modo de sincronización para la voz 3.

**VTYPE 1, NOI,RING.**—Selecciona la forma de onda de ruido y el modo de modulación en anillo para la voz 1.

**ENV** número de voz, ataque, decaimiento, sostenimiento, relajación.— Este comando fija la envolvente para cada voz. El parámetro del número de voz debe de ser un número de 1 a 3. Cada uno de los otros cuatro parámetros requiere un número de 0 a 15.

Envolvente es un término empleado para describir la manera en que el volumen se modifica a medida que una sola nota se toca en instrumentos diferentes. Cada nota, en cuanto al volumen concierne, consiste en cuatro partes básicas: ataque, decaimiento, sostenimiento y relajación. En el momento de tocar una nota, se eleva desde cero hasta un punto máximo de volumen. Esto se llama Ataque, pudiéndose fijar la velocidad con la que esto ocurre. Un valor de 0 significa que el volumen tarda dos milisegundos en ascender desde 0 a su punto máximo, mientras que un valor de 15 significa que tarda ocho segundos. Para los valores intermedios, además de los datos correspondientes para las velocidades de decaimiento y relajación tenéis que consultar las tablas que aparecen en la guía de referencia del programador.

Al llegar a su punto máximo, el volumen cae a algún valor intermedio. La velocidad que emplea para hacer esto se llama el decaimiento. Después del decaimiento viene el sostenimiento. A diferencia de los otros tres parámetros, al valor del sostenimiento no implica un límite de tiempo; indica la proporción de volumen máximo que será sostenida mientras se fija el "gate bit".

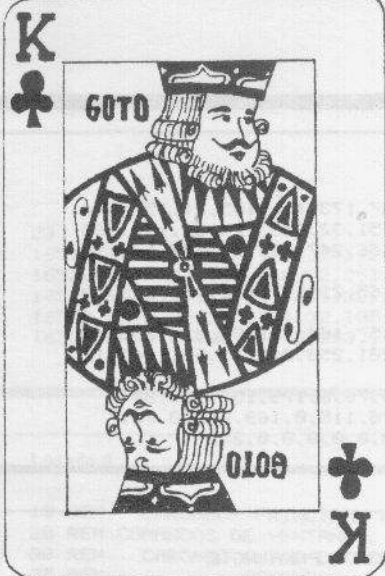
Una vez activado el "gate bit" la voz produce sonido. Cuando se desactiva, se inicia la porción de relajación de la envolvente. Un valor de 15 significa que el volumen máximo será sostenido durante la porción de sostenimiento; un valor de cero, naturalmente, significa que el nivel del volumen es de cero. La velocidad de relajación indica el tiempo que tarda el volumen en caer de su nivel de sostenimiento a cero.

Ejemplo:

**ENV 1,0,9,0,0.**—Fija la voz 1 en una envolvente que produce el sonido de un piano: ataque muy rápido, decaimiento medio, sin sostenimiento ni relajación.

**VOL** valor del volumen.—Fija el volumen, que puede variar de 0, ninguno, a 15, máximo.

**NOTE** número de voz, valor de frecuencia, valor de duración (opcional después un parámetro de cadena como en **VTYPE**).—Este comando toca una sola nota. El número de voz (1-3) debe ser especificado, además del valor de la frecuencia (0-65535). La fórmula para



calcular la frecuencia real en Hz. a partir del valor frecuencia es:

**Frecuencia real** = [(valor de frecuencia)\*.059604645] Hz.

Consulta la guía del usuario para ver los valores correctos de las notas musicales.

Ejemplo:

**NOTE1,4291,20.**—Toca la nota "C" en la voz 1; envolvente, volumen y tipo de voz deben haberse fijado anteriormente; duración de 20.

**NOTE 2,5103,20,PUL255.**—Toca la nota "D#" utilizando la forma de onda de pulsación con un valor de 255.

**NOTE 3,4547,100,TRI,RING.**—Toca "C#" utilizando la forma de onda de triángulo y la modulación en anillo.

**PLAY** número de voz, dirección de inicio de buffer, longitud del buffer.—**PLAY** se usa para tocar una canción entera. Las notas reales serán tocadas por una rutina en código máquina que se ejecuta por interrupciones 60 veces por segundo. Así se permite tocar una canción mientras se está ejecutando un programa.

El número de voz debe ser de 1 a 3. La dirección de inicio del buffer (cualquier zona de memoria RAM) es la que almacena los datos empleados por el comando **PLAY**. La longitud del buffer es el número de bytes de datos desde la dirección inicial.

Los datos tienen que ser almacenados en un formato específico. La rutina **PLAY** espera dos bytes de valor de frecuencia, seguido de un byte que indica el tiempo de sostenimiento de la nota. Un valor de duración de 60 significa que la nota se sostiene durante un segundo, una nota entera. Un valor de 30 significa medio segundo, etc.

Un valor de frecuencia 0 es un caso especial. Cuando surge, un valor de 0 no se introduce en el registro respectivo del valor de la frecuencia de la voz. Aquí lo único que ocurre es que se limpia el "gate bit". Al borrarse este bit se puede oír la porción de relajación de la envolvente. Esto es necesario para ciertos tipos de envolventes en las que la relajación es una parte importante. Entre los programas de demostración, encontra-

rás uno que utiliza el comando **PLAY** para tocar una canción popular.

Ejemplo:

**PLAY 2,\$C000,300:REM COMENZAR A TOCAR EN \$C000 (49152)**

**VIBRATO** número de voz, **ON** u **OFF**.—La rutina **PLAY** es capaz de tocar música en **VIBRATO**, que es una variación de frecuencia muy rápida. A menudo esto produce un sonido más rico, especialmente cuando se está simulando un órgano. Se consigue durante la rutina **PLAY IRQ** al leer el valor actual de la forma de onda de la voz 3 y añadirlo al valor de frecuencia para voz 1, 2, ó 3.

Se requiere un poco de preparación para usar el comando **VIBRATO**. Primero es deseable, aunque no es necesario, apagar la salida de la voz 3. Esto se puede hacer fijando el bit 7 del registro **SID** en la dirección 54296. Fija el volumen con un **Poke** o con **VOL**; luego inicializa la voz 3 con un **Poke**.

El comando **NOTE** no se puede utilizar para iniciar la voz 3 porque borra el "gate bit" automáticamente al terminar, y el "gate bit" tiene que ser fijado para que pueda salir la voz 3.

Ejemplo:

**POKE 54296,128 + 15, REM FIJAR VOLUMEN, APAGAR SALIDA VOZ 3**

**NOTE 3,2000,1:REM FIJAR FRECUENCIA DE LA VOZ 3 ENV 3,0,0,15,0:REM FIJAR**

**POKE \$D412,\$41:REM FORMA DE ONDA DE TRIANGULO DE LA VOZ 3**

**VIBRATO 1 ON**

**PLAY 1,\$C000,200: REM TOCAR CANCION**

#### El tema de los errores

¡Ya tenemos la solución! nuestros hermanos de **RUN** han publicado por fin las correcciones para tres fallos que tenía el **Basic 4.5**:

1. Se quedaba "colgado" al pulsar **RESTORE** a secas.
2. Al hacer **LIST** entre dos números de línea, no paraba al acabar.
3. El comando vibrato no funcionaba (ese problema no lo tendreis vosotros, porque la corrección la teneis aquí).

Hay tan sólo que cambiar las siguientes líneas del programa principal (listado 1 del primer artículo):

```
365 DATA 149,107,137,255,151,71,
    254,1118
595 DATA 24,109,27,212,145,250,200,
    967.
1230 DATA 200,177,95,157,21,208,4,
    902.
```

#### Consejos y advertencias finales

Si has tecleado todos los listados del **Basic 4.5** (incluyendo los demás artículos) verás que la forma de leerlos uno por uno es bastante lenta. Si quieres

**L**os comandos musicales son mucho más fáciles de utilizar que los **POKEs** que normalmente hay que hacer. En primer lugar se define la voz con **VTYP** y **ENV**. Después puede hacerse sonar música con **NOTE** y **PLAY**. Este último comando funciona por interrupciones y no interfiere el funcionamiento de los programas en **BASIC**.

tener en un solo fichero todos los comandos del **Basic 4.5** haz lo siguiente:

Carga y ejecuta todos los listados. Comienza por el "cargador principal" del primer artículo y sigue con los cargadores principales para cada tipo de comandos. A continuación carga y ejecuta todos los comandos por separado. ¡Asegúrate de haber hecho las correcciones!

Una vez hecho esto tienes en la memoria todos los comandos del **Basic 4.5**. Teclea lo siguiente en modo directo:

```
SYS 64738 (conecta el Basic)
BSAVE"BASIC 4.5", PS8000,
PSA000
```

Si tienes unidad de discos pon ,8 después del nombre. Cuando quieras leer el **Basic 4.5** teclea:

```
LOAD "BASIC 4.5",8,1
```

Si no tienes unidad de discos pon un 1 en vez de un 8. Para conectar el **Basic** teclea **SYS 64738** (o pulsa el botón de **Reset**). El nuevo **Basic** queda de este modo conectado. Puedes utilizar el nuevo corrector de listados "Perfecto" para introducir los programas de demostración. Has de cargarlo después de inicializar el **Basic 4.5**.

Como verás, los listados de los comandos no llevan este nuevo tipo de suma de control. Como son todos **DATAs**, con suma de control en cada línea, publicar también otra suma de control sería una pérdida de espacio. El ordenador te dirá si los **DATAs** que has tecleado son correctos o no. Si te sale un error del tipo "Illegal quantity" teclea en modo directo **PRINT L** para saber en qué línea está el error. Seguramente es que te has dejado algún **DATA** o hay alguno cuyo valor es mayor que 255. Corrige la línea correspondiente y graba la nueva versión.



**J**unto con este artículo incluimos varios programas demostrando el uso de ventanas y los comandos musicales. Si el autor no se ha expresado claramente, el estudio cuidadoso de éstos debe subsanar este defecto.

También publicamos los cargadores Basic para los comandos de música y ventanas. Si estás tecleando los comandos de ventanas el cargador que contiene las rutinas principales de ventanas debe ser cargado primero, ya que contiene el código máquina común para todos los comandos de ventanas. Igualmente, si estás tecleando los comandos musicales, el cargador principal para música debe cargarse primero.

Asegúrate de cargar primero el listado 1 del primer artículo de la serie, pues sin él no funciona ninguno de los comandos del Basic 4.5.

#### Listado 1

```
10 REM CARGADOR PRINCIPAL DE MUSICA
20 REM DEBE SER EJECUTADO ANTES DE LOS
30 REM DEMAS COMANDOS DE MUSICA
40 L=14000
50 FORI=37811T037831STEP7:L=L+5:S=0
60 FORJ=0T06:READA:POKEI+J,A:S=S+A:NEXT
70 READSC:IFSC<0:THENPRINT"ERROR EN DATA
S LINEA":L:STOP
80 NEXT
85 :
90 L=14500
100 FORI=37986T038083STEP7:L=L+5:S=0
110 FORJ=0T06:READA:POKEI+J,A:S=S+A:NEXT
120 READSC:IFSC<0:THENPRINT"ERROR EN DATA
S LINEA":L:STOP
130 NEXT
14000 :
14005 DATA 132,151,32,147,130,164,151,90
7
14010 DATA 165,20,32,98,148,200,165,828
14015 DATA 21,32,98,148,76,121,0,496
14500 :
14505 DATA 145,193,145,195,145,195,96,11
14
14510 DATA 32,176,148,138,32,124,148,798
14515 DATA 133,151,32,253,174,32,176,951
14520 DATA 148,138,5,151,96,10,10,558
14525 DATA 10,10,96,32,158,183,138,627
14530 DATA 41,3,170,133,247,202,169,965
14535 DATA 0,202,48,5,24,185,7,391
14540 DATA 208,248,133,195,24,105,189,11
02
14545 DATA 133,193,169,151,105,0,133,884
14550 DATA 194,169,212,133,196,166,247,1
317
14555 DATA 208,3,76,72,178,76,121,734
14560 DATA 0,32,158,183,138,41,15,567
14565 DATA 170,76,121,0,32,121,0,528
14570 DATA 208,3,104,104,96,76,253,844
```

#### Listado 2

```
10 REM CARGADOR PARA PLAY Y VIBRATO
40 L=15000
50 FORI=38085T038176STEP7:L=L+5:S=0
60 FORJ=0T06:READA:POKEI+J,A:S=S+A:NEXT
70 READSC:IFSC<0:THENPRINT"ERROR EN DATA
S LINEA":L:STOP
80 NEXT
15000 :
15005 DATA 201,253,208,19,169,0,160,1010
15010 DATA 28,153,0,212,153,189,151,886
15015 DATA 153,165,151,136,16,244,76,941
15020 DATA 115,0,32,129,148,32,253,709
15025 DATA 174,32,147,130,166,247,165,10
```

61

```
15030 DATA 20,157,173,151,165,21,157,844
15035 DATA 176,151,32,253,174,32,147,965
15040 DATA 130,166,247,165,20,157,167,10
52
15045 DATA 151,165,21,157,170,151,169,98
4
15050 DATA 128,157,164,151,96,32,129,857
15055 DATA 148,201,253,240,13,201,145,12
01
15060 DATA 240,3,76,8,175,157,195,844
15065 DATA 151,76,115,0,169,0,240,751
15070 DATA 246,0,0,0,0,0,0,246
```

#### Listado 3

```
10 REM CARGADOR PARA NOTE
40 L=15500
50 FORI=37751T037810STEP7:L=L+5:S=0
60 FORJ=0T06:READA:POKEI+J,A:S=S+A:NEXT
70 READSC:IFSC<0:THENPRINT"ERROR EN DATA
S LINEA":L:STOP
80 NEXT
15500 :
15505 DATA 32,129,148,32,253,174,160,928
15510 DATA 0,32,179,147,32,253,174,817
15515 DATA 32,158,183,134,151,201,44,903
15520 DATA 208,3,32,203,147,160,4,757
15525 DATA 177,193,9,1,32,100,148,660
15530 DATA 164,151,162,0,32,41,148,698
15535 DATA 32,41,148,202,208,247,136,101
4
15540 DATA 208,242,160,4,177,193,41,1025
15545 DATA 254,76,98,148,132,151,32,891
```

#### Listado 4

```
10 REM CARGADOR PARA VTYPE
40 L=16000
50 FORI=37832T037940STEP7:L=L+5:S=0
60 FORJ=0T06:READA:POKEI+J,A:S=S+A:NEXT
70 READSC:IFSC<0:THENPRINT"ERROR EN DATA
S LINEA":L:STOP
80 NEXT
16000 :
16005 DATA 32,129,148,32,253,174,72,840
16010 DATA 169,0,160,4,32,98,148,611
16015 DATA 32,45,148,104,170,169,16,684
16020 DATA 224,84,240,34,10,224,83,899
16025 DATA 240,29,10,224,80,208,11,802
16030 DATA 32,34,148,160,2,32,179,587
16035 DATA 147,76,6,148,10,224,78,689
16040 DATA 240,3,76,8,175,72,32,606
16045 DATA 42,148,104,32,34,148,32,540
16050 DATA 186,148,201,82,208,9,32,866
16055 DATA 32,148,32,48,148,32,186,626
16060 DATA 148,201,83,208,225,32,48,945
16065 DATA 148,169,2,44,169,4,160,696
16070 DATA 4,17,193,32,98,148,96,568
16075 DATA 160,2,44,160,3,44,160,573
16080 DATA 4,76,251,168,32,176,148,855
```

#### Listado 5

```
10 REM CARGADOR PARA VOL Y ENV
40 L=16500
50 FORI=37941T037985STEP7:L=L+5:S=0
60 FORJ=0T06:READA:POKEI+J,A:S=S+A:NEXT
70 READSC:IFSC<0:THENPRINT"ERROR EN DATA
S LINEA":L:STOP
80 NEXT
16500 :
16505 DATA 32,176,148,173,213,151,41,934
16510 DATA 240,142,213,151,13,213,151,11
```

23

```
16515 DATA 141,213,151,141,24,212,96,978
16520 DATA 32,129,148,32,253,174,32,800
16525 DATA 105,148,160,5,32,100,148,698
16530 DATA 32,253,174,32,105,148,160,904
16535 DATA 6,208,2,145,193,145,195,894
```

#### Listado 6

```
10 REM CARGADOR PRINCIPAL PARA LOS
20 REM COMANDOS DE VENTANAS. DEBE SER
30 REM CARGADO ANTES DE LOS DEMAS
35 REM COMANDOS
40 L=18500
50 FORI=36779T036909STEP7:L=L+5:S=0
60 FORJ=0T06:READA:POKEI+J,A:S=S+A:NEXT
70 READSC:IFS<>SCTHENPRINT"ERROR EN DATA
S LINEA":L:STOP
80 NEXT
85 :
90 L=18600
100 FORI=37171T037188STEP7:L=L+5:S=0
110 FORJ=0T06:READA:POKEI+J,A:S=S+A:NEXT
120 READSC:IFS<>SCTHENPRINT"ERROR EN DATA
S LINEA":L:STOP
130 NEXT
135 :
140 L=18700
150 FORI=37212T037324STEP7:L=L+5:S=0
160 FORJ=0T06:READA:POKEI+J,A:S=S+A:NEXT
170 READSC:IFS<>SCTHENPRINT"ERROR EN DATA
S LINEA":L:STOP
180 NEXT
18500 :
18505 DATA 162,0,134,189,32,237,143,897
18510 DATA 32,106,145,164,173,136,16,772
18515 DATA 5,32,125,145,144,246,165,862
18520 DATA 177,36,189,112,12,169,128,823
18525 DATA 36,189,16,4,81,20,16,362
18530 DATA 2,17,20,145,20,36,189,429
18535 DATA 112,3,76,183,143,165,20,702
18540 DATA 133,36,165,21,41,3,9,408
18545 DATA 216,133,37,165,166,145,36,898
18550 DATA 76,183,143,160,3,132,187,884
18555 DATA 32,158,183,164,187,150,172,10
46
18560 DATA 198,187,48,6,32,253,174,898
18565 DATA 76,241,143,169,0,132,176,937
18570 DATA 173,136,2,133,21,165,175,805
18575 DATA 133,20,24,181,173,176,4,631
18580 DATA 201,41,144,3,76,72,178,715
18585 DATA 165,172,240,12,101,174,176,10
40
18590 DATA 245,201,26,176,241,165,173,12
27
18595 DATA 208,2,104,104,96,32,237,783
18600 :
18605 DATA 164,190,200,162,0,32,68,816
18610 DATA 145,32,68,145,202,208,247,104
7
18615 DATA 136,208,242,96,0,0,0,682
18700 :
18705 DATA 165,20,56,233,40,133,34,681
18710 DATA 165,21,233,0,133,35,96,683
18715 DATA 166,174,202,48,13,165,20,788
18720 DATA 24,105,40,133,20,144,2,468
18725 DATA 230,21,208,240,96,162,1,958
18730 DATA 32,188,145,230,176,165,176,10
32
18735 DATA 197,172,144,2,104,104,96,819
18740 DATA 32,186,145,32,158,183,134,870
18745 DATA 53,32,253,174,32,158,183,885
18750 DATA 134,54,173,32,208,72,165,838
18755 DATA 53,141,32,208,32,51,145,662
18760 DATA 165,54,141,32,208,32,51,683
18765 DATA 145,198,191,208,236,104,141,1
223
```

```
18770 DATA 32,208,96,32,158,183,138,847
18775 DATA 9,1,133,191,32,253,174,793
18780 DATA 32,158,183,134,190,76,253,102
6
18785 DATA 174,0,0,0,0,0,0,174
```

#### Listado 7

```
10 REM CARGADOR PARA RIGHTW Y LEFTW
40 L=17000
50 FORI=36910T036978STEP7:L=L+5:S=0
60 FORJ=0T06:READA:POKEI+J,A:S=S+A:NEXT
70 READSC:IFS<>SCTHENPRINT"ERROR EN DATA
S LINEA":L:STOP
80 NEXT
17000 :
17005 DATA 32,237,143,32,106,145,164,859
17010 DATA 173,192,2,144,243,136,136,102
6
17015 DATA 16,10,200,169,32,145,20,592
17020 DATA 32,125,145,144,236,177,20,879
17025 DATA 200,145,20,76,58,144,32,675
17030 DATA 237,143,32,106,145,165,173,10
01
17035 DATA 201,2,144,207,160,1,177,892
17040 DATA 20,136,145,20,200,200,196,917
17045 DATA 173,144,245,169,32,136,145,10
44
17050 DATA 20,32,125,145,144,233,32,731
```

#### Listado 8

```
10 REM CARGADOR PARA UPW Y DOWNW
40 L=17500
50 FORI=36979T037110STEP7:L=L+5:S=0
60 FORJ=0T06:READA:POKEI+J,A:S=S+A:NEXT
70 READSC:IFS<>SCTHENPRINT"ERROR EN DATA
S LINEA":L:STOP
80 NEXT
17500 :
17505 DATA 32,237,143,198,172,240,110,11
32
17510 DATA 32,106,145,162,1,32,108,586
17515 DATA 145,32,92,145,164,173,136,887
17520 DATA 48,7,177,20,145,34,76,587
17525 DATA 135,144,230,176,165,176,197,1
223
17530 DATA 172,144,12,164,173,136,48,849
17535 DATA 74,169,32,145,20,76,155,671
17540 DATA 144,165,20,133,34,165,21,682
17545 DATA 133,35,162,1,32,108,145,616
17550 DATA 76,133,144,32,237,143,198,963
17555 DATA 172,208,1,96,32,106,145,768
17560 DATA 166,172,32,108,145,32,92,747
17565 DATA 145,164,173,136,48,7,177,858
17570 DATA 34,145,20,76,202,144,230,851
17575 DATA 176,165,176,197,172,144,13,10
43
17580 DATA 164,173,169,32,136,48,5,727
17585 DATA 145,34,76,224,144,96,165,884
17590 DATA 34,133,20,165,35,133,21,541
17595 DATA 32,92,145,76,200,144,32,721
```

#### Listado 9

```
10 REM CARGADOR PARA CENTRE, INV,
20 REM FLASH Y FILL
40 L=18000
50 FORI=37111T037211STEP7:L=L+5:S=0
60 FORJ=0T06:READA:POKEI+J,A:S=S+A:NEXT
70 READSC:IFS<>SCTHENPRINT"ERROR EN DATA
S LINEA":L:STOP
80 NEXT
18000 :
18005 DATA 32,158,173,32,163,182,201,941
18010 DATA 40,144,3,76,154,170,133,720
```



```

18015 DATA 151,169,40,56,229,151,74,870
18020 DATA 166,214,168,24,32,240,255,109
9
19025 DATA 165,151,76,154,170,169,128,10
13
18030 DATA 133,189,76,175,143,32,186,934
18035 DATA 145,32,24,145,32,51,145,574
18040 DATA 32,2,144,32,178,143,198,729
18045 DATA 191,208,243,96,164,190,200,12
92
18050 DATA 162,0,32,68,145,32,68,507
18055 DATA 145,202,208,247,136,208,242,1
388
18060 DATA 96,32,158,183,134,177,32,812
18065 DATA 253,174,32,158,183,134,166,11
00
18070 DATA 169,64,133,189,32,253,174,101
4
18075 DATA 76,175,143,165,20,56,233,868

```

#### Listado 10

```

10 REM      CARGADOR PARA LOCATE
40 L=19000
50 FOR I=38275 TO 38302 STEP 7: L=L+5: S=0
60 FOR J=0 TO 6: READ A: POKE I+J, A: S=S+A: NEXT
70 READ SC: IF SC > 0 THEN PRINT "ERROR EN DATA
S LINEA": L: STOP
80 NEXT
19000 :
19005 DATA 32,158,183,224,40,144,3,784
19010 DATA 76,72,178,134,187,32,253,932
19015 DATA 174,32,158,183,224,25,176,972
19020 DATA 241,164,187,24,76,240,255,118
7

```

#### Listado 11

```

10 ' DEMOSTRACION DE LA UTILIZACION .180
20 ' DE LOS COMANDOS DE VENTANAS .116
30 ' .186
40 ' COLOCA UNA CADENA HACIENDO SCR .32
ALL
45 ' EN EL INTERIOR DE UNA VENTANA .23
50 A$="ESTE ES UN EJEMPLO DE UNA CA .10
DENA MOVIENDOSE A TRAVES DE LA PANT
ALLA."
60 A$=A$+" TAMBIEN PUEDE UTILIZARSE .254
UN BORDE PARA LA VENTANA."
70 A$=A$+" LA CADENA PUEDE SALIR NO .218
RMAL O EN VIDEO INVERSO."
80 A$=A$+" LA VENTANA PUEDE MOVERSE .62
TAMBIEN PARPADEANDO, CON EL COMAND
O INV."
90 ' .246
100 PRINT CHR$(147) .132
110 ' RELLENA UN AREA DE LA PANTALL .96
A CON UN COLOR.
120 FILL 32,1,2,1,36,3 .54
130 ' .30
135 ' .35
136 ' MOSTRAR EN MODO NORMAL .198
140 FOR LOOP=1 TO 50 .18
150 LOCATE 36,2: PRINT MID$(A$,LOOP, .210
1)
155 GOSUB 1000: LEFTW 3,2,34,1 .161
160 NEXT LOOP .254
170 ' .70
180 ' .80
190 ' MOSTRAR EN VIDEO INVERSO .12
200 FOR LOOP=51 TO 100 .58
210 LOCATE 36,2: PRINT MID$(A$,LOOP, .14
1)
220 GOSUB 1000: LEFTW 3,2,34,1 .226
230 NEXT LOOP .68
240 ' .140
250 ' .150
251 ' DIBUJAR UN BORDE .173
252 REV 2,1,36,3: INV 3,2,34,1 .86

```

```

270 FOR LOOP=101 TO 150 .5
280 LOCATE 36,2: PRINT MID$(A$,LOOP, .85
1)
290 GOSUB 1000: LEFTW 3,2,34,1 .41
300 NEXT LOOP .139
310 ' .211
320 ' .221
330 ' INVERTIR LA PANTALLA .175
340 REV 2,1,36,3 .123
350 FOR LOOP=151 TO 190 .225
360 LOCATE 36,2: PRINT MID$(A$,LOOP, .165
1)
370 GOSUB 1000: LEFTW 3,2,34,1 .121
380 NEXT LOOP .229
400 ' .45
410 ' .55
420 ' PANTALLA PARPADEANDO (INV-PEV .229
)
430 FOR LOOP=191 TO LEN(A$) .191
440 LOCATE 36,2: PRINT MID$(A$,LOOP, .245
1)
450 REV 3,2,34,1 : INV 3,2,34,1 .59
460 GOSUB 2000: LEFTW 3,2,34,1 .219
470 INV 3,2,34,1 : GOSUB 2000 .115
480 NEXT LOOP .73
500 ' .145
510 ' .155
520 ' .166
525 FOR T=1 TO 1000: NEXT .21
530 PRINT CHR$(147) .52
990 END .227
1000 FOR PAUSE = 1 TO 50 : NEXT PAU .229
SE : RETURN
2000 FOR PAUSE = 1 TO 30 : NEXT PAU .193
SE : RETURN

```

#### Listado 12

```

10 ' DEMOSTRACION UTILIZANDO LA .20
20 ' PANTALLA ENTERA PARA HACER .56
30 ' SCROLL .24
35 ' .191
40 PRINT "[CLR]"; WID=40: DEP=25 .190
50 DIM A$(94): PRINT "RELLENANDO MATRI .114
Z, ESPERA UN MOMENTO..."
60 ' .216
70 ' RELLENAR ARRAY CON CADENAS DE .180
80 ' 40 CARACTERES QUE FACILMENTE .124
85 ' PODRIAN SER GRAFICOS EN .159
90 ' CUALQUIER JUEGO DE ACCION .16
95 ' .251
100 FOR AR=1 TO 94 .48
110 FOR LO=1 TO 40 .82
120 A$(AR)=A$(AR)+CHR$(AR+161): PRIN .208
TAR,LO "[CRSRL][2SPC][CPSRU]"
130 NEXT LO,AR .226
160 FILL 32,1,0,0,40,25 .94
170 FOR NU=1 TO 4: FOR LO=1 TO 94 .6
180 PRINT "[HOM]"; A$(LO); .42
190 DOWNW ,,,,WID,DEF .46
200 NEXT LO,NUM .58

```

#### Listado 13

```

10 ' DEMOSTRACION DEL COMANDO PLAY .178
20 ' .176
30 ' .186
40 ' AJUSTAR EL VOLUMEN A 15 .182
50 PLAY OFF : VOL 15 .82
60 ' .216
70 ' .226
80 ' LEER DATAS CON LA CANCION Y PO .120
KEARLOS AL BUFFER ($C000)
90 FOR LOOP=0 TO 50 STEP 3 .222
100 READ FR .218
110 HFR=INT (FR/256): LFR=FRAND255 .128
120 POKE $C000+LOOP,LFR .102
130 POKE $C000+LOOP+1,HFR .54
140 READ DUR : POKE $C000+LOOP+2,DUR .190

```

150 NEXT LOOP	.244
160 '	.60
170 '	.70
180 ' ELEGIR VIBRATO O NO VIBRATO	.100
190 PRINTCHR\$(147)"[2CRSRDJ[RVSON]	.176
1 [RVSOFF] TOCAR CON VIBRATO"	
200 PRINT"[CRSRDJ[RVSON] 2 [RVSOFF]	.136
TOCAR SIN VIBRATO[CRSRDJ]"	
210 INPUT:IF A<1 THEN290	.16
220 POKE \$D410,\$8F:"VOZ 3	.154
230 ENV 3,0,0,15,0:"ENVOLVENTE VOZ3	.198
240 NOTE 3,2000,1 :FRECUECIA VOZ3	.246
250 POKE \$D412,\$11:"INCIALIZAR VO	.30
Z 3, ONDA TRIANGULO	
260 '	.161
270 '	.171
280 ' SELECCIONAR ENVOLVENTE	.85
290 ENV 1,0,0,15,0	.61
300 ' SELECCIONAR TIPO DE VOZ	.101
310 VTYPE 1,TRI	.25
320 '	.221
330 ' TOCAR CANCION	.239
340 PLAY 1,\$C000,51	.243
350 '	.251
360 PRINT"[2CRSRDJESTO SIMULA UN OF	.245
GANO."	
370 INPUT"QUIERES OIRLO DE NUEVO":A	.209
\$	
380 IF A\$="S"THEN190	.131

381 '	.26
382 '	.27
383 ' SIMULAR UNA GUITARRA ELECTRIC	.116
A, VOLVER A POKEAR LAS NOTAS	
384 RESTORE:FOR LOOP=0 TO 100 STEP	.111
6	
385 READ FR	.248
386 HFR=INT(FR/256):LFR=FRAND255	.149
387 POKE \$C000+LOOP,LFR:POKE \$C000+	.44
LOOP+3,0	
389 POKE \$C000+LOOF+1,HFR:POKE \$C00	.34
0+LOOP+4,0	
390 READ DUR :POKE \$C000+LOOP+2,DUR	.109
/2 :POKE \$C000+LOOP+5,DUR/2	
391 NEXT LOOP	.230
392 '	.37
393 '	.38
394 ENV 1,0,9,0,0: 'AJUSTAR ENVOLVE	.131
NTE	
395 VTYPE 1, PUL255	.162
396 PLAY 1,\$C000,102	.65
397 PRINT"[2CRSRDJESTO SIMULA UNA G	.196
UITARRA ELECTRIC"	
400 DATA 4817,8,5103,8,5407,8	.63
410 DATA 8583,15,5407,8,8583,15	.171
420 DATA 5407,15,8583,45,9634,8	.21
430 DATA 10207,8,10814,8,8583,8	.5
440 DATA 9634,15,10814,8,8583,8	.75
450 DATA 9634,15,8583,45	.115

VOLUMEN-I 250 Ptas.

# Cursillo de Lenguaje Máquina

BIBLIOTECA  
Commodore  
WORLD

EXCURSION POR LA MICROPROGRAMACION

POR DIEGO ROMERO

## A LA VENTA EN KIOSCOS

Nombre .....  
Dirección .....  
Teléfono .....  
C.P. ....  
Ciudad: .....  
Provincia: .....  
Contrar reembolso .....  
Giro Postal nº .....  
No se aceptan talones.  
Enviar a:  
CW Communications  
C/Barquillo, 21  
28004 MADRID



# Dibujos dinámicos

C-64, C-128 (modo 64) + unidad de discos

por George Trepal

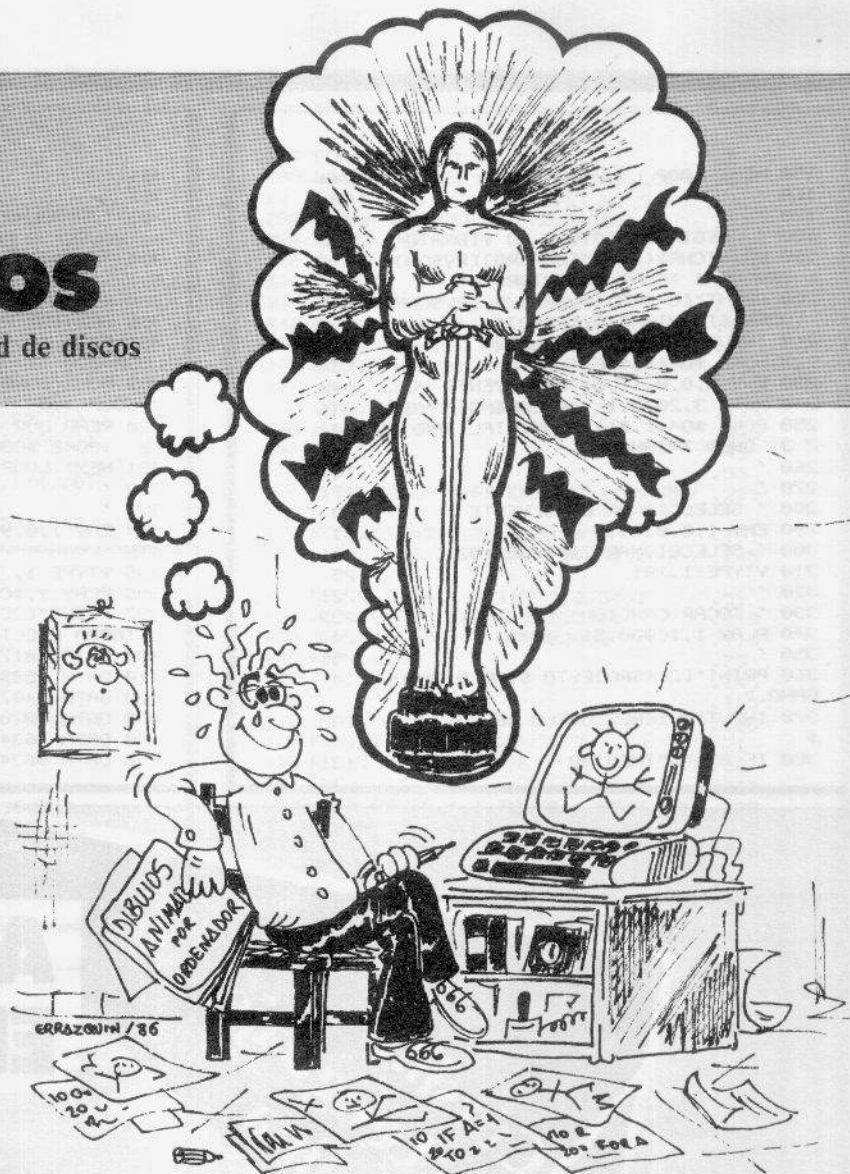
Ahora puedes meter un poco de movimiento en tus presentaciones en pantalla utilizando este programa de gráficos de baja resolución desde el teclado.

**A** diferencia de otros programas de gráficos de teclado convencionales, este produce movimiento, para que puedas disponer de cascadas, cintas transportadoras, ruedas, diseños destellantes y muchos otros efectos. También puedes usar este programa para hacer propaganda y otros anuncios especiales.

Si no has usado nunca este tipo de gráficos, mira el teclado de tu ordenador. La mayoría de las teclas llevan dibujos, o gráficos, incorporados. Para escribir el gráfico de la izquierda, pulsa la tecla Commodore (situada en la parte inferior izquierda del teclado) junto con la tecla deseada. Para escribir el gráfico de la derecha, pulsa la tecla Shift y la tecla deseada. Para cambiar de color, pulsa la tecla CTRL y la tecla de color (fila superior de teclas) elegida. Para pasar al otro juego de colores, pulsa la tecla Commodore y una tecla de color. La tecla 1 es naranja, la 2 es marrón la 3 es rojo claro, la 4 gris oscuro, la 5 gris intermedio, la 6 verde claro, la 7 azul claro y la 8 gris claro.

Puedes hacer combinaciones de estos gráficos para realizar dibujos, bordes y otras cosas. También puedes poner texto en pantalla. La esquina inferior derecha es territorio prohibido. Si escribes un carácter en este sitio, la pantalla hará scroll y se perderá la primera línea de tu dibujo.

Cuando ejecutes el programa, la pantalla quedará en blanco. Los gráficos de teclado funcionan del modo normal, exceptuando los números y algunos signos de puntuación. Al ejecutarlo, el



programa explicará los caracteres que son diferentes. Si pulsas cualquiera de estas teclas modificadas se colocará un carácter con movimiento en la pantalla. Estos caracteres pueden ser combinados con letras y los gráficos normales de teclado para producir dibujos realmente interesantes. No voy a describir la función de cada tecla, ya que quita la emoción de descubrirlo y experimentar.

Prueba esto. Pulsa cualquier tecla de número para formar una fila de imágenes en movimiento. Pulsa la tecla Return para colocar el cursor en el margen izquierdo de la línea siguiente. Tecléa tu nombre y vuelve a pulsar la tecla Return. Ahora pulsa otra tecla numérica para subrayar tu nombre. La tecla "libra" hace parar el programa. Si pulsas las teclas SHIFT y CLR/HOME simultáneamente se borra la pantalla.

Para salvar una pantalla, pulsa la tecla F1, dale un nombre a tu dibujo y pulsa la tecla Return. Cualquier nombre sirve, con la condición de que no exista otro dibujo del mismo nombre en el disco. Se encenderá la luz roja de la unidad de disco, la pantalla dejará de destel-

lar. Cuando vuelva a destellar de nuevo, se habrá salvado la imagen.

Leer una pantalla es casi la misma operación que para cargar un dibujo del disco, excepto que utilizas la tecla F7 en vez de F1.

Al salvar o cargar, se perderá el carácter de la esquina superior izquierda cuando el cursor vuelve a la posición inicial después de acceder a disco.

## Cómo funciona el programa

El programa demuestra un uso divertido de los caracteres personalizados. El ordenador puede realizar millones de operaciones cada segundo pero no hace nada realmente importante. Además, en términos humanos, el ordenador realiza las cosas de la manera más difícil. Por ejemplo, para colocar la letra A en pantalla, tiene que consultar la forma en que se dibuja la A. Después de hacerlo y de dibujar la A, el ordenador olvida lo que acaba de realizar y tiene que volver a repetir el mismo proceso si quieres que te dibuje otra A.

Los caracteres personalizados apa-

• **LOS GRAFICOS EN MOVIMIENTO SE CREAN UTILIZANDO LOS CARACTERES PERSONALIZADOS. TODOS LOS DEMAS CARACTERES GRAFICOS DEL C-64 PUEDEN UTILIZARSE TAMBIEN.**

• **LA PRINCIPAL CARACTERISTICA DEL MOVIMIENTO ES EL SCROLL SUAVE. LAS PRESENTACIONES GANAN CLARIDAD Y COLORIDO.**

• **CON ESTE PROGRAMA PUEDES CREAR UNA PANTALLA DE DIBUJOS Y GRABARLA AL DISCO PARA LEERLA EN OTRO MOMENTO.**

• **TODO EL SECRETO DE LOS GRAFICOS EN MOVIMIENTO ESTA EN LA POSICION DE MEMORIA 53272.**

recen en la pantalla tras diseñarlos e indicarle el ordenador un cambio en el banco de memoria en el que se encuentran los caracteres, que al principio está en ROM. Por lo tanto, aunque el ordenador piensa que está recibiendo órdenes para dibujar una A, realmente le estás indicando que dibuje un marciano, por ejemplo.

Todo este proceso está controlado por el chip VIC-II, que solamente puede trabajar con 16K de memoria a la vez. Dado que un juego de caracteres completo ocupa 2K de memoria, hay espacio para ocho juegos completos. Sin embargo, como se ha utilizado una zona de memoria baja en este programa, sólo hay sitio para seis juegos, ya que gran parte del sistema operativo en Basic y la pantalla ocupan los primeros 4K. Mi programa ocupa los siguientes 4K, lo que deja 8K para los cuatro juegos de caracteres.

El programa realiza el cambio entre los cuatro juegos rápidamente, apareciendo en la pantalla un carácter personalizado diferente cada vez. Igual que los dibujos animados, cada carácter es un poco distinto del anterior, para conseguir la impresión de movimiento. Los caracteres no personalizados son los mismos en cada juego, así que no parece que cambien. El cambio está controlado al decirle a la posición de memoria 53272 el juego de caracteres que tiene que usar.

Cuando se ejecuta el programa, su primera tarea es copiar el juego de

caracteres entero desde ROM a RAM cuatro veces. Después de esto, introduce los caracteres personalizados mediante POKes en los juegos de caracteres de la RAM.

Esto es un programa de demostración, de modo que solamente, se han usado 20 caracteres personalizados. Tu puedes usar muchos más. La ventaja de hacer el cambio de caracteres es que se consigue mucho más movimiento con poco trabajo. Toma nota de que el contenido de casi todo el programa son sentencias data para formar los 20 caracteres. También ocupa sitio el cursor simulado. Para un juego de un laberinto con las paredes electrificadas se podrían construir las paredes con un solo carácter personalizado, haciendo el cambio

entre dos juegos de caracteres solamente. La animación añade un toque diferente a un programa.

Si quieres hacer tu propio programa utilizando mi juego de caracteres, te estarás limitando a 4K de memoria en Basic. Los sprites pueden ser colocados en una parte más alta de la memoria. Parte de mi programa consiste en reservar memoria al hacer un POKE en las posiciones 52 y 56 con el valor 32, colocando los caracteres personalizados en la memoria reservada y cambiando los juegos de caracteres. Este cambio se realiza haciendo un Poke en 53272, (PEEK(53272)AND 240)+X. X se va cambiando el contenido por 8, 10, 12, 14, 8, 10, 12, ...

Para hacer tu propio programa, modifica el mío para poder hacer los POKes con las sentencias DATA en las posiciones correctas. Ejecuta mi programa y luego teclea NEW. El programa se habrá ido pero los caracteres quedarán en su sitio. De esta forma podrás escribir tu propio programa en los 4K de memoria que se han quedado vacíos. Asegúrate de que la primera línea de tu programa reserva la memoria necesaria. Cuando coloques diferentes valores en la dirección mágica 53272, tus dibujos tendrán movimiento.

Si no quieres utilizar mis caracteres personalizados, puedes hacer los tuyos. Consulta "La Guía de Referencia del Programador del C-64" donde encontrarás información sobre caracteres programables.

```
.0
100 REM GRAFICOS EN MOVIMIENTO .190
110 REM PARA EL COMMODORE - 64 .246
120 REM FOR: GEORGE TREPAL .40
130 : .106
140 POKE52,32:POKE56,32:POKE650,128 .4
: DIMTS(39),TC(39):GOTO560
150 REM CURSOR SIMULADO .52
160 PRINT"[CLR]"; .100
170 CD=54272:H=214:L=211 .218
180 P=1024+PEEK(H)*40+PEEK(L)+PEEK(L)>39)*40 .180
190 CH=PEEK(P):K=128 .142
200 POKEP,((CH+K)AND255):POKEP+CD,P .40
EEK(646)
210 POKE53272,(PEEK(53272)AND240)+X .254
220 IFZ$="E" THENPOKE53272,(PEEK(53272)AND240)+4:END .26
```

```
.230 X=X+2:IFX=16THENX=8 .244
240 GETZ$:IFZ$="" THEN200 .234
250 POKEP,CH:PRINTZ$; .236
260 IFZ$=CHR$(34) THENPOKE212,0 .211
270 IFZ$="[SHIFT INST/DEL]" THENPOKE216,0 .217
280 IFZ$="[F1]" THEN320 .207
290 IFZ$="[F7]" THEN450 .119
300 GOTO180 .103
310 REM SALVAR PANTALLA .55
320 FORJ=0TO39:TS(J)=PEEK(1024+J):T .5
C(J)=PEEK(55296+J):POKE(1024+J),32:
NEXT
330 INPUT"[HOM]NOMBRE PROGRAMA";PN$ .173
340 IFPN$="" THEN330 .7
350 PN$=PN$+"$,S,W" .239
360 FORJ=0TO39:POKEJ+1024,TS(J):POK .187
E(J+55296),TC(J):NEXT
```





```

370 OPEN8,8,8,PN$ .205
380 GOSUB1580 .165
390 FORJ=1024TO2023:VL=PEEK(J):PRIN .83
T#8,VL
400 NEXT .155
410 FORJ=55296TO56295:VL=PEEK(J):PR .69
INT#8,VL
420 NEXT:CLOSE8:CLOSE15 .13
430 GOTO170 .217
440 REM LEER PANTALLA .233
450 INPUT"CLRJNOMBRE PROGRAMA";PN$ .45
460 IFPN$="" THEN450 .147
470 PN$=PN$+" ,S,R" .37
480 OPEN8,8,8,PN$ .59
490 GOSUB1580 .19
500 FORJ=1024TO2023:INPUT#8,VL:POKE .95
J,VL
510 NEXT .9
520 FORJ=55296TO56295:INPUT#8,VL:PO .62
KEJ,VL
530 NEXT:CLOSE8:CLOSE15 .124
540 GOTO170 .72
550 REM MOVER JUEGO DE CARACTERES .60
560 POKE53280,12:POKE53281,0:POKE64 .254
6,13
570 PRINT"CLRJ":PRINTSPC(10)"[CRSR .254
DIBUJOS EN MOVIMIENTO"
580 PRINTSPC(12)"[CRSRDIPOR GEORGE .98
TREPAL"
590 PRINTSPC(6)"[2CRSRD]LOS CARACTE .154
RES QUE SE MUEVEN:"
600 PRINTSPC(9)"DE 1 A 9 SIN SHIFT" .140
610 PRINTSPC(11)"6[2SPC]7[2SPC]8[2S .190
PC]9 CON SHIFT"
620 PRINTSPC(6)"Y ,[2SPC].[2SPC]*[2 .70
SPC]+[2SPC]/[2SPC]~"
630 PRINTSPC(2)"[CRSRD]LA E DA POR .198
TERMINADO EL PROGRAMA."
640 PRINTSPC(6)"[CRSRD]F1 SALVA LA .212
PANTALLA AL DISCO"
650 PRINTSPC(6)"[CRSRD]F7 CARGA LA .4
PANTALLA DE DISCO"
660 PRINTSPC(2)"[2CRSRD] LA PACIENC .252
IA ES UNA VIRTUD, ESPERA..."
670 PRINTSPC(3)"[CRSRD]TRASLADANDO .230
JUEGO DE[SHIFT SPC]CARACTERES"
680 J=53248:K=8192:L=10240:M=12288: .90
N=14336
690 POKE56334,PEEK(56334)AND254:REM .198
INTERRUPCIONES APAGADAS
700 POKE1,PEEK(1)AND251:REM I/O OUT .62
710 FORI=0TO2023:P=PEEK(I+J) .244
720 POKEI+K,P:POKEI+L,P:POKEI+M,P:P .202
OKEI+N,P
730 NEXT .230
740 POKE1,PEEK(1)OR4:REM I/O CONECT .172
ADAS
750 POKE56334,PEEK(56334)ORI:REM IN .218
TERRUPCIONES ENCENDIDAS
760 REM INSTALAR JUEGO CARACTERES .168
770 J=8495 .183
780 PRINTSPC(6)"[CRSRD]INSTALANDO J .11
UEGO CARACTERES"
790 J=J+1:READX:IFX<0THEN810 .239
800 POKEJ,X:GOTO790 .93

```

```

810 IFJ<10000THENJ=10543:GOTO790 .33
820 IFJ<12000THENJ=12591:GOTO790 .45
830 IFJ<14000THENJ=14639:GOTO790 .199
840 PRINT"CLRJ[6CRSRD][2SPC]PULSA .1
LA BARRA DE ESPACIO PARA SEGUIR"
850 PRINT"[2SPC]U OTRA TECLA PARA C .229
AMBIAR BORDE"
860 GETA$:IFA$="" THEN860 .71
870 IFA$<>" " THENPOKE53281,(CLAND15 .223
):CL=CL+1:GOTO860
880 X=0:GOTO160 .199
890 DATA0,126,126,96,96,126,126,0,2 .51
4,60
900 DATA102,195,129,24,60,102,24,36 .173
,66,129
910 DATA24,36,66,129,102,60,24,0,19 .103
5,102
920 DATA60,24,129,66,36,24,129,66,3 .107
6,24
930 DATA255,255,0,0,0,255,255,0,255 .13
,0
940 DATA0,0,255,0,0,0,255,255,0,0 .167
950 DATA0,255,255,0,255,0,0,0,255,0 .201
960 DATA0,0,255,129,129,129,129,129 .135
,129,255
970 DATA0,0,0,24,24,0,0,0,192,129 .7
980 DATA3,6,12,24,48,96,3,129,192,9 .135
6
990 DATA48,24,12,6,192,192,192,192, .91
192,192
1000 DATA192,192,3,3,3,3,3,3,3,3 .243
1010 DATA255,255,0,0,0,0,255,255,19 .189
5,195
1020 DATA195,195,195,195,195,195,13 .179
5,36,48,4
1030 DATA108,104,0,49,0,0,0,0,15,15 .72
1040 DATA15,15,204,204,51,51,204,20 .218
4,51,51
1050 DATA-1 .170
1060 DATA0,126,126,102,102,102,102, .14
0,60,102
1070 DATA195,129,24,60,102,195,36,6 .114
6,129,24
1080 DATA36,66,129,24,195,102,60,24 .150
,0,195
1090 DATA102,60,24,129,66,36,24,129 .84
,66,36
1100 DATA255,0,0,0,255,255,0,0,0,0 .116
1110 DATA0,255,0,0,0,255,0,255,255, .30
0
1120 DATA0,0,255,255,0,255,0,0,0,25 .164
5
1130 DATA0,0,0,126,66,66,66,66,126, .134
0
1140 DATA0,0,60,36,36,60,0,0,3,6 .152
1150 DATA12,24,48,96,193,131,192,96 .162
,48,24
1160 DATA12,6,131,193,48,48,48,48,4 .248
8,48
1170 DATA48,48,12,12,12,12,12,12,12 .122
,12
1180 DATA0,255,255,0,0,255,255,0,10 .96
2,102
1190 DATA102,102,102,102,102,102,19 .12
2,196,28,80
1200 DATA0,103,37,32,0,0,0,0,240,24 .12
0
1210 DATA240,240,102,102,153,153,10 .66
2,102,153,153
1220 DATA-1 .84
1230 DATA0,126,126,6,6,126,126,0,10 .154
2,195
1240 DATA0,24,60,102,195,24,66,129, .156
24,36
1250 DATA66,129,24,36,24,195,102,60 .34
,24,0
1260 DATA195,102,36,24,129,66,36,24 .78
,129,66
1270 DATA0,0,0,255,255,0,0,0,0,0 .170
1280 DATA255,0,0,0,255,0,0,255,255, .245
0

```

(pasa a pág. 39)

(viene de pág. 34)



1290 DATA0,0,0,255,0,0,255,0,0,0 .11  
 1300 DATA255,0,0,0,60,36,36,60,0,0 .119  
 1310 DATA0,126,66,66,66,66,126,0,12 .37  
 ,24  
 1320 DATA48,96,192,129,3,6,48,24,12 .209  
 ,6  
 1330 DATA3,129,192,96,12,12,12,12,1 .73  
 2,12  
 1340 DATA12,12,48,48,48,48,48,48,48 .233  
 ,48  
 1350 DATA0,0,0,255,255,0,0,0,24,24 .103  
 1360 DATA24,24,24,24,24,24,0,143,16 .213  
 0,162  
 1370 DATA2,42,200,3,240,240,240,240 .135  
 ,20,20,0,0  
 1380 DATA0,0,51,51,204,204,51,51,20 .91

4,204  
 1390 DATA-1 .255  
 1400 DATA0,102,102,102,102,126,126, .195  
 0,195,0  
 1410 DATA24,60,102,195,24,60,129,24 .31  
 ,36,66  
 1420 DATA129,24,36,66,60,24,195,102 .223  
 ,60,24  
 1430 DATA0,195,66,36,152,129,66,36, .101  
 24,129  
 1440 DATA0,0,255,255,0,0,0,255,0,25 .153  
 5  
 1450 DATA0,0,0,255,0,0,0,255,255 .225  
 1460 DATA0,0,0,0,0,0,0,255,0,0 .217  
 1470 DATA0,255,0,0,0,24,24,0,0,0 .93  
 1480 DATA255,129,129,129,129,129,12 .93  
 9,255,48,96  
 1490 DATA192,129,3,6,12,24,12,6,3,1 .233  
 29  
 1500 DATA192,96,48,24,3,3,3,3,3,3 .245  
 1510 DATA3,3,192,192,192,192,192,19 .19  
 2,192,192  
 1520 DATA0,255,255,0,0,255,255,0,54 .133  
 ,54  
 1530 DATA54,54,54,54,54,54,3,121,0, .179  
 215  
 1540 DATA16,244,6,48,15,15,15,15,0, .198  
 0  
 1550 DATA0,0,102,102,153,153,102,10 .76  
 2,153,153  
 1560 DATA-1 .170  
 1570 REM RUTINA ERRORES .52  
 1580 OPEN15,8,15 .212  
 1590 INPUT#15,E,E\$ .168  
 1600 IFE=0THENRETURN .2  
 1610 PRINTE,E\$;CLOSE8;CLOSE15;END .248



DISTRIBUIDOR EXCLUSIVO

HISPASOFT, S.A.

**Somos una empresa creativa e importadora de informática "Light"**

- Software
- Diskettes
- Accesorios y complementos para micro-ordenadores

## NECESITAMOS

- Distribuidores
- Representantes a comisión
- Colaboradores

## EN TODA ESPAÑA

Para nuestros productos de alta calidad y reciente tecnología

**Escríbanos, indicando su actividad y relaciones en el sector.**



DISTRIBUIDOR EXCLUSIVO

HISPASOFT, S.A.

C/ Coso, 87 - 6ºA - Tfno. (976) 399961 - 500001 - ZARAGOZA



La **MAGIA** son trucos, la **MAGIA** es divertida.

La **MAGIA** es hacer lo que nadie se ha atrevido y resulta ser la fuente más completa de información para la informática práctica.

La **MAGIA** es una sección llena de consejos, trucos, de esto y aquello del mundo del software, hardware y aplicaciones, trucos descubiertos por los demás que hacen que la informática sea más fácil, más divertida o más animada.

**MAGIA** habla de ideas sencillas, programas de una sola línea, subrutinas útiles, hechos de informática poco conocidos y otras cosas de interés.

Los trucos de magia enviárnoslos comprobados, pues hay varios incorrectos.



### AUTO-RUN EN DISCO

Esto es para los fanáticos del AUTO-RUN: Es posible cargar y hacer RUN automáticamente con un programa con el siguiente truco: escribimos LOAD "NOMBRE", 8; posicionamos el cursor justo a la derecha de los dos puntos y entonces apretamos las consabidas teclas SHIFT y RUN/STOP y el programa se cargará y ejecutará igual que si lo cargáramos del cassette.

Luis Martín.  
Matilde Díez, 10  
28002 Madrid.

### CINTAS DE "ACCESO RAPIDO"

Si queremos tener algo equivalente a un directorio en cinta podemos poner al principio de cada cinta el siguiente programa:

```
5 PRINT "(CLR)": POKE 53280,6
10 PRINT "(RVSON) ** DIRECTORIO ** (RVSOFF) (2CRSRD) "
15 LIST 30: END
30 PROGRAMA 1 - BASIC
55 PROGRAMA 2 - (Sys 49152)
70 PROGRAMA 3 - BASIC
79 ** COMIENZO SIGUIENTE PROGRAMA **
```

Los números de línea corresponden a la posición del cuentavueltas del DATTASSETTE, si comenzamos a grabar los programas a partir de la vuelta 30. Así tendremos espacio suficiente para grabar al principio de la cinta el programa DIRECTORIO, que debemos ir actualizando según vayamos metiendo más programas en ella.

Si ponemos la cinta al principio (con el contador en 000) y cargamos el directorio con SHIFT + RUN/STOP, automáticamente veremos un pseudo-directorio de los programas que tenemos en cinta y avanzando hasta la posición que nos indica el falso número podremos cargar directamente el programa que deseemos.

Yo suelo poner el trozo de cinta blanca que indica la unión

entre la cinta magnética y el trozo de cinta plástica que hay al comienzo y final de cada cassette justo encima de la pequeña almohadilla que hay en el centro de la abertura de la cassette y en ese momento pongo el contador a 000. De esa manera tendremos una gran exactitud para poder después localizar los programas.

Normalmente las líneas que hay de la 30 en adelante darían un ?SYNTAX ERROR como un piano, pero como el ordenador nunca las ejecuta (sólo las lista) no nos darán ningún problema.

La última línea es bastante útil ya que nos indica cuál es la posición a partir de la que debemos comenzar el siguiente programa sin tener que hacer el consabido VERIFY "Ultimo programa" y comenzar a grabar a partir del sitio en donde queda la cinta después de verificarlo. Aconsejo que dejéis un par de vueltas "de propina" por lo que pudiera pasar.

Luis Martín  
Matilde Díez, 10  
28002 Madrid

### EL C-16 SE VA A LA GUERRA

Hola, me llamo José Manuel Ruiz Navarro y aquí os envío un truco para la sección de magia. Sirve para hacer que el ordenador suene como unos disparos y luego como una explosión. Estos programas son para el C-16. Este primero es para los disparos y el segundo para la explosión.

```
10 VOL 8: REM PONE EL VOLUMEN AL NIVEL 8
20 FOR S=1000 TO 700 STEP -25
30 SOUND 1,S,1
40 NEXT S
```

```
10 VOL 8
20 FOR S=1000 TO 700 STEP -25
30 SOUND 3,S,3
40 NEXT S
```

José Manuel Ruiz Navarro  
La Palmera, 6, 3.º A  
14006 CORDOBA

## LARGA VIDA AL JUGADOR

Os envío algunos POKES para tener vidas infinitas en algunos juegos:

Blagger 3560,8

Burnin Rubber 18432,173

Choplifter 8011,173

Bruce Lee 5686,128 ó 5677,128

Dig Dug 10473,255

Donkey Kong 12118,238

Dare Devil Denis 29173,255

Falcon Patrol 16764,36:16705,2:SYS6640

Los números sin nada delante son los POKEs.

Ej.: DigDug — 10473,255 es POKE10473,255

*Olaf Scheper  
Turo dien Dent s/n  
Malgrat de Mar  
BARCELONA*

## ARRIBA Y ABAJO

Aquí tenéis dos maneras de subir o bajar una línea de la pantalla sin utilizar ninguna rutina de lenguaje máquina. El scroll hacia arriba se obtiene con:

PRINT"[HOME][25 CRSRD][HOME]"

Y el scroll hacia abajo (sólo en el C-64) puede conseguirse con la siguiente línea:

PRINT"[HOME][CRSRD][CRSRL][SHIFT INST/DEL]";

POKE 218,156

*Shachar Ebel*

## LO NUNCA VISTO

En el Basic del 128, la función MID\$ tiene una nueva característica maravillosa, pero indocumentada: ¡Puedes usar esta función para reemplazar caracteres de una cadena con los de otra! El formato es:

MID\$(cadena 1, posición (,longitud))=cadena 2

Lo que sucede es que la cadena 2 sustituye a los caracteres de la cadena 1 empezando por el carácter indicado en "posición". Lo que queda de la cadena 1 no se ve afectado. La longitud es opcional, pero si no la pones, la cadena 2 no puede ser mayor que el número de caracteres que queden en la primera cadena o aparecerá un ILLEGAL QUANTITY.

*Rick Rothstein*

## DIRECTORIO SELECTIVO

Es bien sabido que puedes cargar una parte determinada del directorio del disco usando algo así como LOAD"\$0:AB\*",8, que carga todos los ficheros que comienzan por AB. Lo que es menos conocido es que puedes utilizar AB\*,CD\* para seleccionar todos los ficheros que comienzan por AB o CD. Puedes utilizar hasta cinco selecciones a la vez y el truco funciona con o sin el Wedge. Las diferentes selecciones que hagas no aparecerán ordenadas en la pantalla, sino en el orden en que están en el disco.

*Lowell K. Unger*

NOTA: Lo que es menos conocido todavía es que se puede utilizar LOAD"\$0:\*=P",8, para listar solamente los ficheros que sean programas, \*=S para los secuenciales, etc. (en vez del asterisco también se puede poner AB\* o cualquier otra cosa). Esto es un pequeño ejemplo sobre la desinformación existente sobre la 1541.



**SEINFO, S.L.**  
SERVICIOS DE INFORMÁTICA

## PROGRAMAS PROFESIONALES

**C Commodore 64 - 128**

### GESTION COMERCIAL

PAQUETE INTEGRADO DE FACTURACION Y CONTROL DE STOCKS

Capacidades de ficheros programables por el usuario (clientes, artículos, proveedores). Control de entradas/salidas de almacén. Inventario permanente e inventario bajo mínimos. Gestión de reserva de pedidos. Facturación y emisión de recibos. Posibilidad de facturar artículos no existentes (facturación directa). Distintos tipos de impuestos programables (ITE, IVA). Listado de entradas/salidas de almacén. Listado de pedidos pendientes, diario de ventas, remesa bancaria. Listados de ficheros con cabecera programable. Listados con criterios de selección de fichas. Emisión de etiquetas. Conexión con tratamiento de textos (documentación personalizada).

### CONTABILIDAD

Basada en el Plan Contable Español. 300 ó 1.000 cuentas. Contrapartida automática. Extractos por pantalla o impresora. Balances programables. Grupos 0 y 9. Balance de situación y cuenta de explotación programables.

### ESTRUCTURAS

Calcula pórticos planos de hormigón armado. Calcula los esfuerzos para las tres hipótesis verticales, viento y sismo. Amado total de vigas y pilares. Cuadro de pesos de hierro. Cuadro cúbico de hormigón. Listado de todos los esfuerzos en el armado.

### MEDICIONES Y PRESUPUESTOS

Programa de mediciones y presupuestos de obra totalmente programable por el usuario. Listado de mediciones y presupuesto por partidas. Posibilidad de ajuste automático de presupuesto.

### FACTURACION

Programa de facturación directa. Fichero de artículos y clientes. Diarios de ventas. Desglose de impuestos. Emisión de recibos. Varias versiones.

### CONTROL DE STOCKS

Ficheros de artículos y proveedores. Control de entrada/salida de almacén. Actualización automática. Inventario permanente. Inventario bajo mínimos. Listados varios.

### DESARROLLOS DE HARDWARE:

- Sistema aula (exámenes tipo test).
- Departamentos de formación.
- Autoescuelas.

### OTROS PROGRAMAS:

- Tratamiento de textos en español.
- Generador de ficheros.
- Estadística (5 paquetes), etc.

## LIDER EN VENTA DE PROGRAMAS PROFESIONALES

Pida información (976) 22 69 74-23 29 61

**SEINFO, S. L.**

Avda. Goya, 8 - 50006 ZARAGOZA



# MARKET CLUB

Servicio gratuito para nuestros lectores **PARTICULARES**. Los anuncios serán publicados durante 1 mes. Los anuncios gratuitos de Market Club **SOLAMENTE** serán publicados si vienen con nombre, apellidos y dirección completa.

## MERCADILLO

- Vendo Commodore 64 + Datassette + 2 Joysticks + 3 manuales + revistas nacionales y extranjeras + programas por 55.000, precio real en el mercado 90.000 Ptas. en perfectas condiciones. Recién revisado por Iberdata, con garantía. Unidad de discos 1541, como nueva y con muchos discos y programas (procesador de texto, base de datos) 45.000 Ptas. Se vende todo o parte de ello por no usarlo. Manuel Giner Nadal, Paseo de la Pechina, 29, Valencia. Teléf.: (96) 370 13 64, entre semana y a la hora de comer. (Ref. M-652).
- Desearía que me pusieran en contacto con algún club de aquí de Zaragoza, para el VIC-20 o en todo caso, alguno para el VIC-20, aunque sea en otra ciudad. Javier Alfranca, C/Teniente Ortizte Zárate, 1, 3.º I, 50015 Zaragoza (Ref. M-653).
- Vendo VIC-20 con superexpander + 3K, ampliación 16K, libros innovative computing, 60 programas For VIC, guía del usuario, muchas revistas y las dos partes del curso Basic (con cintas), todo por 20.000 Ptas. Llamas a: Eusebio Zuloaga Arisa, C/ Balmes, 444, 08022 Barcelona. Teléf.: (93) 247 22 76 (Ref. M-654).
- Vendo Commodore 64 con programas, juegos, lenguajes y utilidades. Precio a convenir. Llamas a: Eusebio Zuloaga Arisa, C/ Balmes, 444, 08022 Barcelona. Teléf.: (93) 247 22 76 (Ref. M-655).
- Estudiante con economía baja, necesita Datassette de 2.ª mano en buen estado. Mandar ofertas a: Juan Carlos Posac Zarza, Tr. Príncipe de Viana, 6, Peñafiel (Valladolid). (Ref. M-656).
- Vendo consola videojuegos Philips con monitor B/N incluido, todo por estrenar, por 18.000 Ptas. Llamas (93) 345 06 15, Quique. Paseo Torras i Bages 81 4-4, 08030 Barcelona, (Ref. M-657).
- Vendo ordenador Commodore-64 y Cassette Commodore en perfecto estado por 50.000 pesetas. Vendo impresora matricial Commodore MPS-801 en muy buen estado por 50.000 Ptas. Resto de facilidades al llamar al (91) 638 54 79 preguntando por JESUS, a partir de las 5 P.M. (Ref. M-658).
- Estoy dispuesto a comprar expansión de 3K para el VIC-20 por el precio de 2.000 Ptas. (DISCUTIBLE). Interesados llamar o escribir a: Pablo Lestau Sáenz, Avda. de la Estación, 7, Calahorra (La Rioja). (Ref. M-659).
- Vendo Commodore 64 con revistas, libros, Datassette, Joystick y programas variados para este ordenador, precio por todo el equipo: 50.000 Ptas. Los interesados llamar al teléfono: 461 72 59 de Bilbao, preguntando por Eduardo o escribir a: Eduardo López García, Apartado de correos n.º 37 Santurce (Vizcaya). (Ref. M-660).
- Urge vender Commodore 64, con garantía Microelectrónica y Control, junto con transformador, cable conexión al televisor, dos joysticks, manual del usuario, los dos tomos del curso de Basic del C-64, 495 juegos y utilidades, y todo ello en perfecto estado, precio a convenir. Interesados dirigirse a: Tony Rueda Roselló, C/Padilla, 216, 1.º, 4.º, 08013 Barcelona. Teléf.: (93) 245 99 38. (Ref. M-661).
- Vendo Commodore Plus/4 de junio 85' + Unid. 1531 todo a 52.000 Ptas. Con garantía, apenas estrenado. Incluyo dos manuales de Basic 3,5 y otro de 4 programas construidos en el mismo micro. 61,8K disponibles. O cambio por C-64 + C2N y una generosa cantidad de software o interfaces. Escribir: Carlos Felipe Spada, C/ Rosellón, 290, 08037 Barcelona o llamar al 257 75 51 mediodía o noche. (Ref. M-662).
- Commodore 128, intercambio para cualquiera de sus tres modos operativos información, programas y experiencias. En definitiva, todo aquello que redunde en un mejor aprovechamiento de esta máquina. Francisco Loperena, C/ Daoiz, 2, Ceuta. Teléf. (956) 51 57 19. (Ref. M-663).
- Vendo videojuegos Atari modelo CX 2600, cinco Joystick, 7 cartuchos de juegos y todos los cables. Todo por 20.000 Ptas. Llamas (93) 894 37 58. Javier Pérez-Ullivarri, C/Balmes, 32, Sitges (Barcelona). (Ref. M-664).
- Se vende commodore VIC-20 precio a convenir según posibilidades económicas alrededor de las 22.000 Ptas. pero incluyendo curso de Basic 1.ª parte, juego original Ratman cartucho de juego de marcianos, cartucho ampliación a 16K y un interface para conectar el ordenador con cualquier cassette (precio real 50.000) ideal para iniciarse en la informática. Escribid a Joan M. Robles Arans C/Dr. Trueta 31, 2.º, 3.ª Castelfelers (Barcelona). (Ref. M-665).
- Vendo curso de introducción al Basic: parte 1 y 2 (de Andrew Colins) y los programas de test ya pasados a cassette. Regalo al que me los compre algunos juegos. Todo por 3.500 Ptas. Escribid a Angel Antonio Francisco Santos C/Rúa Nova 8, 1.º 32004 Orense, o llamar de lunes a jueves y de 10 a 12 al (988) 23 38 01 extensión 21. (Ref. M-666).
- Urgente vendo Vic-20 en perfecto estado, Datassette, (10.900), guía para el usuario del Vic-20 (3.200) y 20 revistas (5.520) todo por 22.000 Ptas. Llamas preferentemente desde Barcelona al Teléf. 332 47 32 de 2 a 3 pm. o de 7 a 9 pm. Enrique Alvarez, Jacinto Verdagué, 13, 5.º, 1.º, Hospitalet (Barcelona) (Ref. M-667).
- Vendo CBM-16, como nuevo (enero-85) con cassette, cables, manuales, etc. El mejor Basic del mercado (do, loop, until, while, circle, else, etc.) + 2 cintas originales, más lote de revistas Commodore. Con monitor de código máquina incluido en Rom. Todo por 38.000 Ptas. Llamas al (927) 41 16 67, o escribir a: Narciso Mesa Bravo; Ronda del Salvador, 85, Plasencia (Cáceres). (Ref. M-668).
- Vendo Commodore 64 (enero 85), impresora MPS-801, 1530 Datassette Unit C2N, joystick 1311 y Paddle 1312, además daría algún juego, las revistas Commodore World. Todo por 90.000 Ptas. Dirigirse a: Fernando Ochoa Equizabal, Avda. Navarra, 69, 5.º B. 50001 Zaragoza. Teléf.: (976) 32 80 00 (Ref. M-669).
- Vendo Joystick para Commodore por 1.700 Ptas. También vendo seis revistas con programa y comentarios sobre Commodore Vic-20 y C-64 —500 Ptas.— todo, Julián Sánchez; Urb. Colombia, C/ Tunaima, casa 2, 5.º dcha. Barrio de Hortaleza. Madrid. Teléf.: 763 49 63 (Ref. M-670).
- Vendo ordenador Vic-20 por 17.500 Ptas. con cartuchos, juegos, manuales, revistas, etc., 27.500 Ptas. Interesados diríjanse a la dirección: Carlos Galán, C/ Mestre Millet, 6, bajo 2.º, Gava (Barcelona) o al teléf.: (93) 62 50 90 y pregunten por Carlos. (Ref. M-671).
- Vendo por 2.000 Ptas. cartuchos Vic-20. Ayuda al programador y Sargon II (ajedrez) o cambio por otros cartuchos. Manuel Antonio Vidal Reynés, C/ Málaga, 37, 6.º, 2.ª 08007 Palma de Mallorca (Balears). Teléf.: 27 74 57 (Ref. M-672).
- Vendo Vic-20 unidad de cassette Commodore, palanca de joystick, libro introducción al Basic 1.ª parte, con cassettes, cartuchos. Aparato TV. Sanyo pequeño en blanco y negro, todo 40.000 Ptas. Teléf.: 323 59 58, Xavi Barceló (Ref. M-673).
- Vendo Commodore 64 con joystick, juegos y libros con menos de un año por 39.000 Ptas. Llamas al (93) 214 31 72 de Barcelona a partir de las 22 horas y preguntar por Jaime (Ref. M-674).
- Vendo Amstrad-664, fósforo verde, cable para cassette, juego ajedrez en disco 3-D voice chess con voz en castellano, dos cintas de juego, manuales y varios disquetes todo por 100.000 Ptas., también vendo impresora Printer-80 Indescomp por 35.000 pesetas. Interesados escribir a: Pedro Márquez Jiménez, C/ Velarde, 11, 2-B. Linares (Jaén). (Ref. M-675).
- Vendo impresora MPS-801 poco uso (por adquisición de otro modelo superior) por 43.000 pesetas. José Font Ballester, Mayor, 33; Beniarbeig (Alicante). Teléf.: (965) 76 67 56 (Ref. M-676).
- Compró números en perfecto estado de conservación de: Commodore World: n.º 1 al 10, y 12, 13 y 15. Commodore Magazine: n.º 1 al 5 y el 7 y listados de programas de cualquier tipo, preferentemente música. Cambiaría algunos números por: C-64 Tape Computing (n.º 1), VideoCommodore (n.º 1), y Commodore computer club n.º 1, o los vendería. También cambio juegos (tengo pocos). Ofertas a: Antonio Saá Quintas, C/ El Almandro, 120, Aguadulce, (Almería). Teléf. (951) 34 02 93. De 16,00 a 20,00. (Ref. M-677).
- Vendo Commodore Vic-20 comprado hace un año, junto con cartuchos de 16K y ROAD ROCE y 50 juegos entre los cuales hay varios de la revista Commodore World, todo por 20.000 Ptas. Interesados dirigirse a: Luis Cerón Naharro, calle Plaza de España, 5, 2.º, Almedrales (Badajoz). Teléf.: (924) 66 36 88 (Ref. M-678).
- Por cambio de ordenador, a modelo más potente, vendo: ordenador Vic-20 (15.000 Ptas.) comprado en julio de 1984, Datassette (8.000 Ptas.) comprado en agosto 85, ampliación de memoria de 64Kb. "Ferre Moret", amplia Rom, Eprom, Ram, por (15.000 Ptas.). Lote de 250 programas, los regalo con la compra del ordenador. Si se compran varios objetos, precio especial. Revista: Commodore World. Commodore Magazine. Daniel Roig Marchuet; C/ Castilla, 27 bajos, Ibiza (Balears) (Ref. M-679).
- Por cambio de ordenador, vendo Vic-20, Datassette, manual de usuario y revistas. Todo por 18.000 Ptas. Vicente Berengüé Roca, C/ Alfonso el Sabio, 38, 9.º A. Alicante. Teléfono: 21 96 89. (Ref. M-680).
- A quien le pueda interesar otrezo una impresora prácticamente nueva (muy poco uso) MPS-801 COMMODORE. El precio es de unas 45.000 Ptas., lo que yo pido. Con la impresora obsequiaré con el programa EASY SCRIPT. Ramón Petit Torrent, C/ Feixa Llarga, s/n. L'Hospitalet, Barcelona. Teléf.: 336 26 39 (Ref. M-681).
- Vendo impresora MPS-801 para Commodore, con once meses, por tener otra. Estado impecable, 45.000 Ptas. Carlos José García Carratala, Urb. Tres encinas, chalet, 59, Valdepastores-Las Lomas Boadilla del Monte (Madrid). Teléf.: 410 25 86 (Ref. M-682).
- Vendo CBM-64, unidad de discos, impresora Riteman C+, Joysticks, libro introducción al Basic, discos y los mejores programas del mercado. Comprado noviembre. Garantía oficial. Precio a convenir. José Marsá, C/ Prats y Roque, 32, entr. 1.º, 08027 Barcelona. Teléf.: 352 98 90 (Ref. M-683).
- Vendo ordenador Commodore Vic-20 + Superexpander (con 3K incluidos) + I y II parte de la introducción al lenguaje de programación Basic + Vic-20 Guía del usuario (Osborne/McGraw-Hill). Todo por 13.500 Ptas. Dirigirse a Javier Torres de Zubizarreta, Rambla de Méndez Núñez, 45, 03002 Alicante. Teléf.: (965) 21 20 32 (Ref. M-684).
- Por compra de otro ordenador, regalo MPS-801 con garantía por 25.000 Ptas. y C-64 por 25.000 Ptas. Todo junto: 40.000 Ptas. Juan Rafael Oscar Martín Mihalic; Guayadeque, 4, 3.º Izda. 35009 Las Palmas de Gran Canaria. Teléf.: (928) 26 24 81 (Ref. M-685).
- Vendo Vic-20+Datassette+cables+manual+introducción al Basic I+cartucho Vic-Avenger+cintas vírgenes+revistas. Todo en buen estado por 16.000 Ptas. Toni Farras Pique, Ctra. de Lérida, sin número, Orgaño (Lérida) (Ref. M-686).
- Desearía vender un Commodore 64, en perfecto estado, incluyendo unos 200 programas, utilidades, juegos y música, Simon'basic y también el manual del usuario en español. Todo

esto por el precio de 50.000 ptas. Interesados, ponerse en contacto con: Miguel Angel Rico Duro. C/ Mallorca, 28, 8.º 1.º. 43005 Tarragona. Teléf.: (977) 22 37 03 (Ref. M-687).

• Se vende unidad de disco 1541 en 40.000 ptas. Interesados llamar al teléf. 25 84 77 y preguntar por Emilio. Sólo residentes en la isla de Gran Canaria. Precio fijo. Emilio Torres González. Avda. Escaleritas, 70 2.º E, Las Palmas de Gran Canaria (Ref. M-688).

• Vendo por razones puramente económicas ordenador Vic-20 más Datassette, cartuchos, tres libros de programación, varios juegos en cinta. Todo por 20.000 ptas. Llamar al teléfono 219 72 07 de Barcelona, preguntar por Manolo, me urge vender rápido. Manuel Flores Pérez. C/ Tenerife, 27 entr. 1.º. 08024 Barcelona (Ref. M-689).

• Vendo Commodore 64 e impresora 801, por 40.000 ptas., cada uno. Antonio Recober Giner. Barriada Parque Mediterráneo, bloque 1, 71. 29004 Málaga. Teléf.: (952) 32 20 46. (Ref. M-690).

• Vendo C-64 totalmente nuevo de trínca (sin sacar ni siquiera de la caja) junto a los correspondientes libros, datassette y joystick, también sin estrenar. Daniel Garriga. C. N. 2, 2-51; San Just Desvern (Barcelona). (Ref. M-691).

• Compró Vic-20 que esté en buen estado, con Datassette. Sin ampliaciones de memoria por 13.000 ptas. Javier Moscardó Llaró. C/ Cuatretonda, 6. Beniganim (Valencia). Teléf.: 221 58 54 (Ref. M-692).

• Vendo ordenador Commodore SX-64 (portátil, con unidad de discos y monitor color incluido) por importe de 150.000 ptas. Regalo 100 programas de juegos y utilidades en disco. Dirigirse a: Adolfo Martínez Pérez; Carmen, 6, 5.º A. Elche (Alicante). (Ref. M-693).

## TRABAJO

• Diseñador de portadas.—Se ofrece para cassettes de ordenador. Tel.: (91) 661 77 61, o bien, Manuel Alonso Expósito, c/ Dr. Ignacio Barraquer, 2, 3º 2º Sant Boi (Barcelona). El precio será a convenir.

• Imparto clases particulares de iniciación a la informática, metodología de la programación, programación en lenguaje Basic, lenguaje Logo para niños. Con prácticas en microordenador Commodore. Horario a convenir. Información en el teléfono: 27 69 78 de Granada. Miguel Angel.

• Clases particulares de Informática. Conoce a fondo tu Commodore 64. Hardware y Software. Escribir a: Josemaría Canals. Pintor Soler Jorba, 4-5-2. Cerdanyola (Barcelona).

• Se ofrece abogado-contable, licenciado en Derecho (1978-1983). Especialista en contabilidad e impuestos. Esteban T. Noci Muñoz. Paseo San Illán, 57. 28019 Madrid. Tel.: 469 67 44.

• Se busca alguien para impartir clases particulares sobre proceso de textos (Easy Script) en el C-128 —es el mismo que el del C-64—. También sobre la conexión de órganos a través del interface MIDI. Llamar al teléfono 733 18 97 de Madrid. Preguntar por Dra. Sousa.

## CLUBS

• Quisiera contactar con usuarios de Commodore-64, para formar Club en Badalona. Se podrían compartir muchas experiencias. Contactar con: Pedro Buenaventura Llanos. Avda. Martín Pujol, 298, 1.º-2.º. Badalona (Barcelona). Tel. 389 52 96. (Ref. C-87).

• Calc Result y Superbase. Soy usuario de ambos programas. Intercambio experiencias; CR no lo domino, SB la conozco bastante. Josep Rovira i Sardá. Cavallers, 17, 2.º-1.º. Sant Sadurní d'Onoia (Barcelona). Tel. (93) 891 07 40. (Ref. C-88).

• Me gustaría ponerme en contacto con algún usuario de Practicale 64. José Ignacio Vicario López. C/ Martínez de la Riva, 11. 28018 Madrid. Teléfono (91) 478 90 03. (Ref. C-89).

• Este apartado está dedicado a los usuarios del C-16, para que demos lo que este modelo es capaz de hacer. Es una asociación llamada MULTI-SOFT 16, sin ánimo lucrativo los interesados pueden escribir a: Alfonso García Tejero, bloques Ciudad de Málaga, bloque 1.º-2.º. C/ Melilla. O bien pueden llamar al número: (952) 68 42 70 a partir de las 6,30 de la tarde. Cada uno hará una parte del programa. (Ref. C-90).

• Club (C) LDA. Lleida. Intercambiamos experiencias y lo que se presente (programas, hard, etc.) con otros clubs y con particulares. Andreu Ibáñez i Perales. C/ Passeig de Ronda, 76, 3.º-1.º. 25006 Lérida. (Ref. C-91).

• ¡Atención alquimistas del sonido! Club de información y software para sintetizadores. Solicita información a: ADSR síntesis. Gavá, 71 B, 2.º-3.º. 08014 Barcelona. (Ref. C-92).

• Club de programas para Commodore-64. Interesados escribir a Manuel Arroyo. c/ Maresma, 266-2º-2.º. 08020 Barcelona. (Ref. C-93).

• Deseo contactar con un club de usuarios de C-64 en Madrid. Escribir a: Francisco Javier Ruiz. C/ Hogar de Belén, 84. 28037 Madrid. (Ref. C-94).

• Toda persona interesada en programar en Forth, intercambiar experiencias en este lenguaje, o simplemente ayuda puede ponerse en contacto con nosotros. Escribir a: Manuel Fuentes Sorriás, Grupo Forth Galicia. Apartado de Correos, 1473 Vigo (Pontevedra). (Ref. C-95).

• Se ha creado CO64 (Club de Ordenadores C-64). Nuestros objetivos son muy variados, desde intercambio de programas hasta la compra venta de Hardware. Interesados escribir a Joan Carles Amador, C/ Esteve Vila, 1, Salt (Girona), o a Narcís Patiño, C/ Enderrocades, 11, 17004 Girona. (Ref. C-96).

• Club CBM Rosas: abierto a todos los usuarios del CBM-64 intercambiamos juegos, ideas, experiencias y programas de toda clase escribir a Joaquín Llanas. C/ Eugeni d'Ors nº 3-2º Rosas (Gerona). (Ref. C-97).

• Descarta inscribirme a algún Club de Vic-20, sin fines lucrativos, para intercambio de programas y otras cosas. Tengo bastantes programas. Interesados escribir a: David Noviembre Naranjo. C/ Santiago, 44. Hinojos (Huelva). (Ref. C-98).

• APUPO (Asociación Palenciana de Usuarios de Ordenadores Personales). Deseamos entrar en contacto con otros clubs e intercambiar programas para el Commodore 64 (disco y cinta) y Spectrum (cinta). Contactar con Pablo Acinos; Panaderos, 14, 5.º C. 34001 Palencia (Ref. C-99).

• ¿Te aburres con tu C-64? ¿Ya no sabes cómo sacarle partido? Pensamos que aún no lo has intentado todo. Hemos formado en Huelva el primer club para usuarios de C-64 y C-128 de capital y provincia. Estamos dispuestos a intercambiar todo tipo de programas, sin fines lucrativos. Nuestra programoteca recoge más de 500 títulos. ¡Vamos, animate! Llama al (955) 24 66 48 o escríbenos a: COMMONUBA COMPUTER CLUB. C/ Rafael Guillén, 1, 3.º C. 21002 Huelva (Ref. C-100).

• Atención comodóneros y spectronianos vamos a crear en nuestra provincia un Club de usuarios de Commodore 64 y Spectrum. Fines: intercambio de ideas, programas, ayudas, cursillos y concursos. Estamos en trámites de editar una revista. Creación de un juego-club. Para información: Miguel Angel de Rueda. Apdo. 935. 04080 Almería o al teléfono 34 02 93. Preguntar por Antonio Saá de 16.00 a 20.00. (Ref. C-101).

### DESEAN CONTACTAR CON OTROS AMIGOS COMMODORIANOS

#### C-16

• Iñaki García Rodríguez. C/ Urbitarte, 6, 4.º B. Azpeitia (Guipúzcoa). Poseo cassette.  
• José Manuel Ruiz Navarro. C/ La Palmera, 6, 3.º A. Tel. (957) 27 24 38. 14006 Córdoba. Poseo cassette.

#### VIC-20

• José García Rodríguez. C/ Cosme Andrade, 23. Tel.: 40 04 82. Ponferrada (León). Poseo cassette.  
• René Suárez. Tiraña. Laviana (Asturias). Poseo cassette.  
• Francisco Parrilla Lucena. C/ Generalísimo, 1. Tel. 74 06 55. Baeza (Jaén). Poseo cassette.  
• Francisco Martínez Villalba. C/ P. de Rivera, 19. 1.º, 3.º. Tel.: 229 31 94. Beniganim (Valencia). Poseo cassette.

#### C-64

• Jordi Malé Vila. Ausias Marc, 4, 2.º, 3.º. Blanes (Gerona). Poseo cassette.  
• José Tomás Cuello. Arriba, 9, 11, 5.º. Tel.: (923) 21 84 18. 37002 Salamanca. Poseo cassette y unidad de disco.  
• Luis Miguel Montero. Alejandro Sánchez, 8, 2.º A. Tel.: (91) 471 14 28. 28019 Madrid. Poseo cassette y unidad de disco.  
• Pedro Alcántara Enríquez. Pol. Virgen Africa, 6, 6.º B. Ceuta. Poseo unidad de disco.  
• José Luis Gallardo García. C/ Circunvalación, 2, 1.º C. Tel.: 11 98 20. 18007 Granada. Poseo cassette.  
• Alejandro Pariente Ruiz. C/ Eugenio Gross, 50, 1.º C. 29009 Málaga. Poseo cassette, 300 programas.  
• Jorge Mansa Sánchez. P.º de Ronda, 36, 4.º. 25003 Lérida. Poseo cassette.  
• Matéu Pastoret Jou. C/ F. Cambó, 3. Tel.: 59 02 08. Besalú (Gerona). Poseo cassette.  
• Pablo Fortea Ortiz. C/ Portugal, 24, 5.º Dcha. Tel.: (965) 22 77 66. 03003 Alicante. Poseo cassette y unidad de disco.  
• Pedro García de Paredes. C/ Sonora, 3. Tel.: 80 23 90. Hernán Cortés (Badajoz). Poseo cassette.  
• Antonio Francisco Saá Quintas. C/ El Almendro, 120. Tel.: (951) 34 02 93. Aguadulce (Almería). Poseo cassette.  
• Manuel González Lombardía. C/ Gabriel y Galán, 4, 2.º A. Peñaranda de Bracamonte (Salamanca). Poseo cassette y unidad de disco.  
• Manuel Jesús Álvarez Márquez. P.º de Independencia, 25, 4.º A. 21002 Huelva. Tel. 25 42 46.  
• Juan Carlos Balbas Pérez. Daoiz y Velarde, 21, 1.º Dcha. Tel.: (942) 21 68 19. 39003 Santander. Poseo cassette y unidad de disco.  
• Francesc Planas Mateu. Apdo. de Correos, 1. Tel. 36 47 67. Lloret de Mar (Gerona). Poseo cassette y unidad de disco.

#### C-128

• Eduardo González Granda. Plaza Fonsagrada, 5, 10.º, pta. 4. 28029 Madrid. Tel.: 201 91 18. Poseo unidad de disco.

Deseo contactar con otros amigos Commodorianos

Nombre .....

Dirección.....

Telf.: ..... Ciudad: .....

C.P. .... Provincia .....

Modelo de ordenador .....

Tengo Cassette ..... ☐

Unidad de Disco ..... ☐





## COLABORACIONES

"Colaboraciones" es vuestra sección que consiste en los programas que vosotros nos mandáis. Todos los meses elegiremos la mejor colaboración de acuerdo con su calidad, utilidad y originalidad, y su autor recibirá un premio de 5.000 pesetas. Los programas deben ser enviados en cinta o en disco (que luego serán devueltos a su propietario) así como una explicación breve del programa y un listado del mismo. Es importante incluir el nombre, dirección y teléfono del autor (estos datos no serán publicados si así lo desea el interesado).



### Master Music

C-64, C-128 + Simon's Basic

José Luis Moreno Vigil  
C/ Gravina, 34. at. 1.  
Hospitalet (Barcelona)

Es un sintetizador que nos permite modular cualquiera de las tres voces del 64 e incluso utilizarlas

todas a la vez. (sonido polifónico).

El manejo es muy sencillo. Basta con pulsar la tecla en inverso para acudir a esa opción.

Este programa nos permite investigar sonidos y el modo de ejecutarlos. Utilizando la polifonía, los filtros (hay tres tipos 1, 2, 3 que son pasa altos, pasa bajos y pasa banda. O nos devuelve a la normalidad) se pueden conseguir efectos muy interesantes.

Las opciones que preguntan ...alto, ...bajo se refieren a byte alto y byte bajo.

Para utilizar la forma de onda-pulso, hay que definir anteriormente su longitud (ancho del pulso).

Para experimentar los efectos polifónicos, basta con dejar un sonido y cambiar de voz. El sonido que hemos dejado pasará a ser el sonido de fondo.

Espero poder mandar próximamente la continuación de este programa, que ahora estoy ultimando. Se trata de un editor musical, un pentagrama que nos permitirá escribir cualquier canción utilizando los sonidos definidos en esta primera parte. Ambos pueden utilizarse unidos o por separado, puesto que son independientes.



```
1 COLOUR 0,0:PRINT"[VEL]":HIPES 0 .43
1:TEXT 0,0,"MASTER MUSIC",1,2,8
2 TEXT 55,80,"JOSE LUIS MORENO",1, .202
3,12
3 TEXT 150,160,"(C) COLOCON[2SPC]1 .77
985",1,2,8:BLOCK 40,60,260,130,2
```

```
4 PAUSE 10:NRM .12
5 FORI=54272T054296:POKEI,0:NEXT .249
10 DIMA%(3),B%(3),S%(3),R%(3),P(3): .68
S=SOUND :V=1:P(1)=26:P(2)=29:P(3)=3
2:O%=3
20 A%=RND(0)*15+1:FORI=1T03:A%(I)=A .244
%:D%(I)=A%:S%(I)=A%:R%(I)=A%:NEXT:G
OSUB110
30 FORI=15T021STEP2:PRINTAT( P(1),I .234
)"A%AT( P(2),I)"A%AT( P(3),I)"A%
:NEXT
40 DIM A$(185),IN%(3),O$(4),F%(3),F .20
(3):VOL 15:IN%(1)=4:IN%(2)=11:IN%(
3)=18
43 F(1)=31:F(2)=47:F(3)=79:F(4)=15 .113
45 O$(1)="F1":O$(2)="F3":O$(3)="F5" .241
:O$(4)="F7"
50 C$="Q2W3ER5T6YU":B$="C[SHIFTC]D .92
[SHIFTD]JEF[SHIFTF]G[SHIFTG]A[SHIFTA
]B"
60 FORI=1TOLEN(C$):A=ASC(MID$(C$,I, .180
1)):A$(A)=MID$(B$,I,1):NEXT
70 GETA$:IFA$=""THEN70 .114
80 IF A$<"2"ORFA$<"[F2]"THEN70 .56
90 IFA$<"Y"ANDFA$<"[F1]"THEN70 .178
100 CGOTO ASC(A$)*10 .190
110 PRINT"[CLR][CRSR][RVSON][2SPC] .236
[2CRSR][2CRSR][SHIFT-][2CRSR]
[2CRSR][2CRSR][SHIFTB]"
130 PRINT"[CRSR][RVSON][2SPC][RVSO .100
FF]2 [RVSON] [RVSOFF]3 [RVSON] [SHI
FT-] [RVSOFF]5 [RVSON] [RVSOFF]6 [R
VSON] [RVSOFF]7 [RVSON] [SHIFT-]"
140 PRINT"[CRSR][RVSON][2SPC][2CRS .30
RR] [2CRSR] [SHIFT-] [2CRSR] [2CR
SR] [2CRSR] [SHIFT-]"
150 PRINT"[CRSR][RVSON][2SPC][2CRS .40
RR] [2CRSR] [SHIFT-] [2CRSR] [2CR
SR] [2CRSR] [SHIFT-]"
160 PRINT"[CRSR][RVSON][2SPC][COMM .130
M][2SPC][COMM][2SPC][SHIFT-] [COMM
M][2SPC][COMM][2SPC][COMM][2SPC][
SHIFT-]"
170 PRINT"[CRSR][RVSON] Q[COMM] W .240
[COMM] E[SHIFT-]R[COMM] T[COMM]
Y[COMM] U[SHIFT-]"
180 PRINT"[CRSR][RVSON][2SPC][COMM .150
M][2SPC][COMM][2SPC][SHIFT-] [COMM
M][2SPC][COMM][2SPC][COMM][2SPC][
SHIFT-]"
190 PRINTAT( 22,1)"OCTAVA :":O%AT( .240
23,2)"[RVSON]:[RVSOFF][2SPC][RVSON]
:[RVSOFF]"AT( 22,4)"F1[5SPC]: TRIAN
GULO"
200 PRINTAT( 22,5)"F3[5SPC]: SIERRA .92
"AT( 22,6)"F5[5SPC]: PULSO"
210 PRINTAT( 22,7)"F7[5SPC]: RUIDO .122
"AT( 22,9)"[RVSON]:[RVSOFF]10Z EN CU
R50:"
215 PRINTAT( 22,8)"F2[5SPC]: OFF VO .121
Z "
```

```

220 PRINTAT< 1,13>"ONDA[9SPC]-"AT< .212
1,15>"[RVSON]A[RVSOFF]TAQUE[7SPC]:"
230 PRINTAT< 1,17>"[RVSON]D[RVSOFF] .48
ECAIMIENTO[2SPC]:"AT< 1,19>"[RVSON]
S[RVSOFF]OSTENIMIENTO:"
240 PRINTAT< 1,21>"RELA[RVSON]J[RV .196
OFF]ACION[3SPC]:"AT< 1,23>"[RVSON]P
[RVSOFF]ULSO[8SPC]:"
250 PRINTAT< 25,10>"[COMMJ][9COMMJ] .238
[COMMJ]"AT< 25,11>"[RVSON][COMMJ]E
VSOFF] 1[2SPC]2[2SPC]3 [COMMJ]"
260 FOR I=12022:PRINTAT< 25,I>"[RV .161
ON][COMMJ][RVSOFF][9SPC][COMMJ]":NE
XT
270 PRINTAT< 25,23>"[COMMJ][RVSON]I .55
9COMMJ][RVSOFF][COMMJ]"AT< 1,8>"[RV
SON]F[RVSOFF]ILTRO[8SPC]:"AT< 1,9>"
[RVSON]L[RVSOFF]ONGITUD ="
280 RETURN .83
300 E$=" [CLR]" + RIGHT$(STR$(V),1)+A$ .111
(ASC(A$))+RIGHT$(STR$(O$),1)+"[F5]"
310 MUSIC 10,E$:PLAY 2:GOTO70
500 GOTO300 .191
510 GOTO300 .201
520 GOTO70 .238
530 GOTO300 .222
540 GOTO300 .232
550 GOTO300 .242
560 GOTO70 .22
570 GOTO70 .32
580 OZ=OZ-1:OZ=OZ-(OZ<0):PRINTAT< 3 .166
0,1>"":OZ:GOTO70
590 OZ=OZ+1:OZ=OZ+(OZ>6):PRINTAT< 3 .8
0,1>"":OZ:GOTO70
600 GOTO70 .62
610 GOTO70 .72
620 GOTO70 .82
630 GOTO70 .92
640 GOTO70 .102
650 PRINTAT< 16,15>"":FETCH "[CRSR .244
DJ",2,RZ(V):IFRZ(V)<00RZ(V)>15THEN
650

```

```

655 GOSUB1400:PRINTAT< P(V),15>"[3S .207
PC][3CRSRL]":RZ(V):GOTO70
660 GOTO70 .122
670 GOTO70 .132
680 PRINTAT< 16,17>"":FETCH "[CRSR .30
DJ",2,DZ(V):IFDZ(V)<00DZ(V)>15THEN
680
685 GOSUB1400:PRINTAT< P(V),17>"[3S .33
PC][3CRSRL]":DZ(V):GOTO70
690 GOTO300 .126
700 PRINTAT< 16,8>"":FETCH "[CRSRD .44
J",1,FZ(V):IFFZ(V)<00RFZ(V)>3THEN70
0
705 IFFZ(V)=0THENPOKES+23,0:VOL 15 .179
:ELSE :VOL F(FZ(V)):POKES+23,2*(V-
1)
706 GOTO70 .168
710 GOTO70 .172
720 GOTO70 .182
730 GOTO70 .192
740 PRINTAT< 16,21>"":FETCH "[CRSP .214
DJ",2,RZ(V):IFRZ(V)<00RRZ(V)>15THEN
740
745 GOSUB1400:PRINTAT< P(V),21>"[3S .51
PC][3CRSRL]":RZ(V):GOTO70
750 GOTO70 .212
760 PRINTAT< 10,9>"BAJA ":FETCH " .186
[CRSPDJ",1,LBZ:IFLBZ<00RLBZ>7THEN76
0
761 PRINTAT< 10,9>"ALTA ":FETCH " .113
[CRSPDJ",3,LAZ:IFLAZ<00RLAZ>255THEN
761
765 POKES+21,LBZ:POKES+22,LAZ:GOTO7 .181
0
770 GOTO70 .233
780 GOTO70 .243
790 GOTO70 .253
800 PRINTAT< 9,23>"ALTO ":FETCH " .227
[CRSPDJ",2,PAZ:IFPAZ<00PPAZ>15THEN8
00
801 PRINTAT< 9,23>"BAJO ":FETCH " .32
[CRSPDJ",3,PBZ:IFPBZ<00PPBZ>255THEN

```

## GLOSARIO

**Fetch - Captura o recogida.**—Ciclo o instrucción que localiza y toma datos de un sistema de almacenamiento. Generalmente se emplea para definir la parte del ciclo de ejecución en que la CPU direcciona la siguiente posición del contador de programa y lee el dato o instrucción contenido en ella.

**File - Archivo.**—Conjunto de datos o registros relacionados entre sí y tratados como una unidad lógica.

**Fill character - Carácter de relleno.**—Aquel utilizado para ocupar una zona vacía en la pantalla, en un fichero o en impresora.

**Firmware.**—Componentes de un ordenador que no se pueden denominar como Software o Hardware; por ejemplo: Un cartucho ROM que contenga algún programa, o una unidad de disco inteligente que contiene sus propios programas de funcionamiento, sistema operativo, etc.

**Fixed point.**—Punto decimal.

**Flag - Bandera.**—Señalizador o indicador de que algo ha ocurrido.

**Flicker - Centelleo.**—El parpadeo indeseable en las pantallas del monitor o pantalla del ordenador. Es muy perjudicial para la vida.

**Floppy disk - Disquete.**—Disco de material magnético depositado sobre una base plástica, se emplea para el almacenamiento de datos.

**Flowchart - Diagrama de flujo.**—Representación gráfica del análisis, definición, programas, procedimientos de resolución de problemas, etc.

**Flyback - Retorno.**—Este término se emplea para referirse al trazo (no visible en la pantalla) recorrido por el haz de electrones desde el ángulo inferior derecho hasta el superior izquierdo del tubo de rayos catódicos.

**Font (character font) - Tipo de caracteres.**—Un conjunto de caracteres que tiene un estilo y tamaño determinado.

**Foreground - Primer plano.**—En sistemas multitarea o multiprograma es el entorno en que el programa de mayor prioridad se ejecuta.

**Form - Formulario.**—El papel sobre el que la impresora imprime los caracteres.

**Format - Formato.**—La distribución de datos en el medio sobre el que se almacenan.



```

801
805 POKES+INX(V)-1,PAZ:POKES+INX(V) .254
-2,PBZ:GOTO70
810 GOTO300 .247
820 GOTO300 .1
830 PRINTAT( 16,19)";:FETCH "[CPSR .193
D1";2,SZ(V):IFS(V)<00FSZ(V)>15THEN
830
835 GOSUB1400:PRINTAT( P(V),19)"[3S .174
PC][CPSRL]";SZ(V):GOTO70
840 GOTO300 .21
850 GOTO300 .31
860 PRINTAT( 36,9)";:FETCH "[CPSRD .159
1";1,V:IFY<10PV>3THEN860
865 ENVELOPE V,AZ(V),DZ(V),SZ(V),R .122
Z(V):GOTO70
870 GOTO300 .51

```

```

880 GOTO70 .87
890 GOTO300 .71
1330 WAVE V,00010000:GOSUB1400:PRI .231
NTAT( P(V)+1,13)"[2SPC][2CRSRL]";0$
(1):GOTO70
1340 WAVE V,00100000:GOSUB1400:PRI .1
NTAT( P(V)+1,13)"[2SPC][2CRSRL]";0$
(2):GOTO70
1350 WAVE V,01000000:GOSUB1400:PRI .153
NTAT( P(V)+1,13)"[2SPC][2CRSRL]";0$
(3):GOTO70
1360 WAVE V,10000000:GOSUB1400:PRI .115
NTAT( P(V)+1,13)"[2SPC][2CRSRL]";0$
(4):GOTO70
1370 WAVE V,00000000:GOTO70 .249
1400 ENVELOPE V,AZ(V),DZ(V),SZ(V), .9
RZ(V):RETURN

```

## Cuestionario

Vic-20 sin expansión

José Abraham Parra Cordero  
C/ Aneto núm. 2-1º A  
Cáceres

Os envío dos programas para el Vic-20, uno que título Fichero Cuestionario "Ficuestionario", y

que especialmente está indicado para aquellos que están estudiando o están preparando oposiciones, pues este programa os puede ayudar mucho en vuestros estudios ya que con él podréis haceros vuestros propios cuestionarios con tres supuestos y de la materia que queráis, y como es de suponer hay que averiguar cuál de los tres supuestos es el correcto.

El problema que se me presentaba cuando decidí hacer un programa de este tipo era que, al ser mi ordenador un Vic-20 sin expansión,



disponía de muy poca memoria, y si ponía el cuestionario en DATAS o en un fichero en el que el ordenador cargara todo el cuestionario en memoria, las preguntas que me cogían eran poquísimas. Por eso decidí hacerlo tipo fichero, pero sin que se cargue en memoria todo el fichero, es decir, ir visualizando las preguntas a medida que las va cogiendo el cassette. Y en memoria, la siguiente pregunta que sustituya a la anterior. Únicamente se cargan en memoria todas las respuestas.

De esta forma y tal y como está hecho el programa, puedes hacerte tu cuestionario de hasta 100 preguntas, que es la cantidad de que suelen constar los cuestionarios de las oposiciones a Funcionarios del Estado.

La descripción del programa de una manera simplificada es la siguiente:

De la línea 10 a la 85. Visualizan un cuadro con el nombre del programa y de su autor.

86-96. Visualizan un menú con sólo dos opciones; la opción número uno es para cargar datos (grabar en cassette el cuestionario de preguntas y respuestas).

La opción número dos es para lectura de datos (ir cogiendo del cassette las preguntas con los tres supuestos y esperar a que el usuario conteste cuál es el correcto).

100-224. Visualizan las normas para la grabación de respuestas y preguntas, y que son:

Primero se graban las respuestas correctas y por tanto tendrás que ir pulsando el 1, el 2, o el 3, según que el supuesto correcto sea uno, dos o tres.

Estos números se irán cargando sin pulsar nunca return y en el orden en que luego cargarás las preguntas y supuestos.

Cuando hayas terminado de cargar todas las respuestas correctas, pulsarás un número mayor de tres y entonces podrás comenzar a grabar las preguntas con los tres supuestos, y cuyas normas son:

—Tampoco pulsarás nunca return.

—Para hacer punto y aparte, lo señalarás con un asterisco (\*)

—Cuando hayas puesto el último supuesto, el tercero, lo señalarás con el signo (#).

—Cuando pongas la última pregunta con sus tres supuestos y después de poner el signo (#) tendrás que pulsar además simultáneamente las teclas SHIFT y O.

226-290. Abren fichero cuestionario para cargar datos en él y grabar las respuestas correctas.

300-360. Graban las preguntas con sus tres supuestos.

380. Ordena que comience de nuevo el programa en la línea 86 (visualización del menú nuevamente).

500-570. Extrae todas las respuestas correctas y las carga en memoria.

580-650. Van extrayendo las preguntas y los tres supuestos, carácter a carácter, y las visualiza en pantalla mediante la subrutina 1000-1130, y espera a que el usuario teclee cuál de los tres supuestos es el correcto para comprobar si es el verdadero, función que realiza otra subrutina 800-880.

670-700. Visualiza en pantalla el nº de preguntas acertadas y el nº de preguntas formuladas.

800-880. Subrutina que comprueba la respuesta del usuario y que es llamada desde la otra subrutina que a continuación describo.

1000-1130. Subrutina que imprime en pantalla las preguntas y los tres supuestos sin que queden cortadas las palabras.

```

10 PRINTCHR$(147):POKE36879,8:DIMB(101)
20 PRINT"[WHT] *****"
30 PRINT" *[18SPC]*":PRINT" *[3SPC]CUEST
IONARIO[3SPC]*"
40 PRINT" *[3SPC][12COMMU][3SPC]*"
50 PRINT" *[4SPC]POR ABRAHAM[3SPC]*"
60 PRINT" *[18SPC]*":PRINT" *[7SPC]PARRA
[6SPC]*"
65 PRINT" *[18SPC]*":PRINT" *[3SPC]C/ ANE
TO,2-1-A[2SPC]*"
70 PRINT" *[18SPC]*":PRINT" *[5SPC]CACER
ES[6SPC]*"
75 PRINT" *[18SPC]*":PRINT" *****
*****"

```

```

80 PRINT:PRINTTAB(4)"[RVSON]PULSA UNA TE
CLA"
85 GETC$:IFC$=""THEN85
86 PRINTCHR$(147):PRINTTAB(8)"MENU":PRIN
TTAB(8)"[4COMMJ]":PRINT:PRINT
87 PRINTTAB(4)"[RVSON]1[RVSOFF] CARGAR D
ATOS":PRINT:PRINTTAB(4)"[RVSON]2[RVSOFF]
LECTURA DATOS"
90 GETC$:C=VAL(C$)
95 IFC<10RC>2THEN90
96 ONCGOTO100,500
100 PRINTCHR$(147)
110 PRINT"[6SPC]NORMAS":PRINT"[6SPC]6CO
MMJ]"
130 PRINT:PRINT"[RVSON]1[RVSOFF]-ENTRADA
RESPUESTAS
140 PRINT"[2SPC]CORRECTAS:"
150 PRINT:PRINT"[4SPC]UNA TRAS OTRA":PRI
NT"[4SPC]SIN PULSAR RETURN"
156 PRINT"[4SPC]TERMINAR CON UN":PRINT"[
4SPC]NUMERO MAYOR DE 3"
160 PRINT:PRINT"[RVSON]2[RVSOFF]-ENTRADA
PREGUNTAS:"
170 PRINT:PRINT"[2SPC]NUNCA PULSAR RETUR
N":PRINT:PRINT"PUNTO Y APARTE CON *"
190 PRINT:PRINT"FIN PREGUNTA CON #":PRIN
T:PRINT"FIN CUESTIONARIO [SHIFT0]"
210 PRINT:PRINT"[RVSON]1[4SPC]PULSA UNA T
ECLA[RVSOFF]"
220 GETC$:IFC$=""THEN220
224 PRINTCHR$(147)
226 OPEN1,1,2,"CUESTIONARIO"
230 PRINT"ENTRADA DE RESPUESTAS"
240 PRINT:PRINT"CORRECTAS:"
250 GETA$:IFA$=""THEN250
260 PRINTA$:
270 PRINT#1,A$:
280 IFASC(A$)<49ORASC(A$)>51THEN300
290 GOTO250
300 PRINTCHR$(147):PRINT"[WHT]ENTRADA DE
PREGUNTAS:"
310 PRINT
320 GETA$:IFA$=""THEN320
330 PRINT#1,A$:
340 PRINTA$:
350 IFA$="[SHIFT0]"THEN370
360 GOTO320
370 CLOSE1
380 GOTO96
500 PRINTCHR$(147):M=1
520 OPEN1,1,0,"CUESTIONARIO"

```

```

530 GET#1,B$
540 B(M)=VAL(B$)
550 IFB(M)>3THEN580
560 M=M+1
570 GOTO530
580 PRINTCHR$(147):M=0
590 GET#1,A$
630 A1$=A1$+A$
634 A=LEN(A1$)
640 IFA=23THENGOSUB1000
645 IFA$="[SHIFT0]"THENGOSUB1000:GOTO660
650 GOTO590
660 PRINTCHR$(147)
670 PRINT"[3SPC]HAS ACERTADO ";L
680 PRINT:PRINT"[3SPC]DE ";M;" PREGUNTAS
"
690 PRINT:PRINT"[3SPC]FORMULADOR."
700 END
800 REM COMPRUEBA RESPUESTAS
810 GETR$:IFR$=""THEN810
820 R=VAL(R$)
830 IFR=B(M)THENL=L+1:PRINT:PRINT"MUY BI
EN":GOTO860
840 PRINT:PRINT"MAL, LA RESPUESTA"
850 PRINT:PRINT"CORRECTA ES ";B(M)
860 FORY=1TO1500:NEXTKK
870 PRINTCHR$(147)
880 RETURN
1000 REM IMPRIME CADENA SIN CORTAR PALAB
RAS
1010 AA$=A1$
1020 PP=LEN(AA$)
1025 FORQQ=1TOPP
1030 IFMID$(AA$,QQ,1)="*"THENRR=QQ-1:GOS
UB1080:A1$=RIGHT$(AA$,PP-QQ):RETURN
1031 IFMID$(AA$,QQ,1)="#"THENRR=QQ-1:GOS
UB1080:A1$=RIGHT$(AA$,PP-QQ):M=M+1:GOSUB
800:RETURN
1035 NEXTQQ
1040 FORQQ=PPTO1STEP-1
1050 IFMID$(AA$,QQ,1)=" "THENRR=QQ-1:GOS
UB1080:A1$=RIGHT$(AA$,PP-QQ):RETURN
1060 NEXTQQ
1070 RR=22:GOSUB1080:A1$=RIGHT$(AA$,PP-2
2):RETURN
1080 FORQQ=1TORR
1090 PRINTMID$(AA$,QQ,1);
1095 FORKK=1TO25:NEXTKK
1100 NEXTQQ
1110 PRINT
1120 IFR<22THENPRINT
1130 RETURN

```

## COMMODORE WORLD EN DISCOS

NOMBRE .....  
 DIRECCION .....  
 POBLACION ..... TELEF. ....  
 (.....) PROVINCIA .....

DESEO RECIBIR EL DISCO CON LOS PROGRAMAS DE LA REVISTA N° .....

PRECIO DEL DISCO 2.000 PTAS. — SUSCRIPTORES DE LA REVISTA, 1.750 PTAS.

SOY SUSCRIPTOR ☐ N° DE SUSCRIPTOR .....

DESEO SUSCRIPCION ANUAL (11 DISCOS) A PARTIR DEL N° (Suscripción 17.500 Ptas)\*

☐ Incluyo cheque por valor de ..... pesetas Firma.....  
☐ Envío giro n° ..... por ..... pesetas

SI DESEAS RECIBIR LA REVISTA EN DISCOS PARALELAMENTE A LA EDICION IMPRESA, ENVIANOS ESTE CUPON. EL DISCO **SOLO** LLEVA GRABADOS LOS PROGRAMAS DE LA REVISTA PERO **NO** LOS ARTICULOS. CADA DISCO, **A PARTIR DEL N° 14 INCLUSIVE**, VA EN SU ESTUCHE CON SU PORTADA CORRESPONDIENTE A TODO COLOR.

(\*) La suscripción no puede iniciarse con números anteriores al 14.



# Sopa de letras

C-16

Gonzalo Fernández Val  
Plaza Adriano, 7, 2º 2.<sup>a</sup>  
08021 Barcelona

Mando un juego a su lograda revista inspirado en el programa "sopa de letras" que apareció

en el número 10 de Commodore World.

Paso a explicar las instrucciones del juego.

Al ejecutar el programa, el ordenador pide el tamaño de la matriz.

Introducimos el número de filas y de columnas, el cual deberá estar comprendido entre los valores entre corchetes.

Luego nos pide el número de palabras a buscar (valor máximo entre corchetes).

Pide el tiempo para buscar cada palabra. A continuación nos aparece la matriz y la palabra. Inmediatamente después de imprimirse la matriz y la palabra, suena un pitido, lo que indica que el tiempo empieza ya, hasta que introducimos la fila en que se encuentra la palabra en cuestión. Después introducimos la columna (01-30).



La condición necesaria para poder hacer un récord es encontrar todas las palabras. El ordenador nos muestra en azul las palabras encontradas y en rosa las no encontradas.

Cuando nos pregunta si queremos volver a jugar otra partida, pulsando otra tecla distinta de "N" y "S", el ordenador visualizará la matriz con las palabras al descubierto. Pulsando una tecla cualquiera, se vuelve a la pantalla anterior.

El juego es fácilmente adaptable al C-64. Los POKES de las primeras líneas sirven para guardar los datos relativos al récord, para que cuando hagamos un CLR, el récord permanezca en memoria.

Como se puede ver, el listado es un poco largo, pero está hecho en descomposición arriba-abajo. (no hay subrutinas).

## Estructura del programa

líneas	contenido.
40	almacenamiento récord.
70-240	entrada tamaño de la sopa de letras y número de palabras a buscar.
250-350	asignación palabras data a una tabla.
360	inicio bucle principal (I).
400-660	ordena la palabra en la matriz en sentido horizontal.
670-930	ordena la palabra en la matriz en sentido h. inverso.
940-1200	ordena la palabra en la matriz en sentido vertical.
1210-1460	ordena la palabra en la matriz en sentido v. inverso.
1470	final bucle principal (I).
1710	inicio bucle principal (II).
1720-1960	entrada e impresión de la matriz. (entrada filas/columnas).
1980-2120	cambia el color azul de las palabras a rosa.
2140	final bucle principal (II).
2150-2280	impresión del récord y el tiempo tardado.
3000-3020	líneas data de palabras (modificables).
El ordenador distribuye las palabras en la matriz al azar, pudiéndolas ordenar de 4 modos. Las palabras no se cruzan entre sí. Puede haber como máximo una palabra en cada fila/columna.	

```

10 REM -----
20 REM - SOPA DE LETRAS -
30 REM -----
40 POKE 148,255:POKE 149,255:POKE 150,25
5
50 PRINT "[CLR][11CRSRD]";TAB(11)"* SOPA
  DE LETRAS *":GETKEY A$
60 CLR:PRINT "[CLR][14SPC][COMM0]"
70 PRINT "[2CRSRJENTRADA TAMANO DE LA M
  ATRIZ":PRINT "[CRSRJ[29COMMV]"
80 INPUT "[CRSRJNUMERO DE FILAS [15,20]
  ";R%
90 IF R%<15 OR R%>20 THEN 80
100 INPUT "[CRSRJNUMERO DE COLUMNAS [15
  ,30]";C%
110 IF C%<15 OR C%>30 THEN 100
120 DIM M$(R%,C%),A$(3),R%(2)
130 FOR I=1 TO R%
140 FOR J=1 TO C%
150 M$(I,J)="

```

**Commodore**  
**WORLD**

Barquillo, 21, 3º Izda.  
Teléf.: 231 23 88/95  
28004 MADRID

Sant Gervasi de Cassoles, 39-despacho 4  
Teléf.: 212 73 45 / 212 88 48  
08022 BARCELONA

```

160 NEXT J
170 NEXT I
180 MAX=10
190 IF RZ=20 AND CZ>19 THEN MAX=20
200 PRINT:PRINT:PRINT "[2CRSRR]NUMERO DE
PALABRAS A BUSCAR [1,";RIGHT$(STR$(MAX)
,2);"]]"
210 PRINT "[CRSRR][36COMMV]"
220 INPUT "[CRSRR]PALABRAS";NWZ
230 IF NWZ<1 OR NWZ>MAX THEN 220
240 DIM R1Z(NWZ),C1Z(NWZ),P$(NWZ),M$(NWZ)
)
250 POZ=INT(RND(1)*NWZ)+1
260 FOR I=1 TO POZ
270 READ P$
280 NEXT I
290 P$(1)=P$
300 IF NWZ=1 THEN 360
310 FOR I=2 TO NWZ
320 READ P$
330 IF P$="*" THEN RESTORE:READ P$
340 P$(I)=P$
350 NEXT I
360 FOR I=1 TO NWZ
370 IF LEN(P$(I))>15 OR LEN(P$(I))<1 THE
N PRINT "[CLR]ERROR EN DATAS.":END
380 MZ=INT(RND(1)*4)+1
390 ON MZ GOTO 400,670,940,1210
400 AZ(1)=0:AZ(2)=0:ROZ=INT(RND(1)*RZ)+1
:LPZ=LEN(P$(I)):R1Z(I)=ROZ:DZ=0
410 FOR J=1 TO CZ
420 Z$=M$(ROZ,J)
430 IF Z$="" THEN 470
440 DZ=DZ+1
450 IF DZ>2 THEN 940
460 AZ(DZ)=J
470 NEXT J
480 IF AZ(1)=0 AND AZ(2)=0 THEN P1Z=CZ-L
PZ:DZ=0:GOTO 620
490 IF AZ(2)>0 THEN 550
500 A1Z=AZ(1)-1:A3Z=CZ-AZ(1)
510 IF LPZ>A1Z THEN 530
520 P1Z=A1Z-LPZ:DZ=0:GOTO 620
530 IF LPZ>A3Z THEN 940
540 P1Z=A3Z-LPZ:DZ=AZ(1):GOTO 620
550 A1Z=AZ(1)-1:A2Z=AZ(2)-AZ(1)-1:A3Z=CZ
-AZ(2)
560 IF LPZ>A1Z THEN 580
570 P1Z=A1Z-LPZ:DZ=0:GOTO 620
580 IF LPZ>A2Z THEN 600
590 P1Z=A2Z-LPZ:DZ=AZ(1):GOTO 620
600 IF LPZ>A3Z THEN 940

```

```

610 P1Z=A3Z-LPZ:DZ=AZ(2)
620 POZ=INT(RND(1)*(P1Z+1))+DZ:C1Z(I)=PO
Z+1
630 FOR N=1 TO LPZ
640 A$=MID$(P$(I),N,1):M$(ROZ,POZ+N)="[C
OMM6]" + A$
650 NEXT N:MZ(I)=1
660 GOTO 1470
670 AZ(1)=0:AZ(2)=0:ROZ=INT(RND(1)*RZ)+1
:LPZ=LEN(P$(I)):R1Z(I)=ROZ:DZ=0
680 FOR J=CZ TO 1 STEP -1
690 Z$=M$(ROZ,J)
700 IF Z$="" THEN 740
710 DZ=DZ+1
720 IF DZ>2 THEN 1210
730 AZ(DZ)=J
740 NEXT J
750 IF AZ(1)=0 AND AZ(2)=0 THEN P1Z=CZ-L
PZ:DZ=0:GOTO 890
760 IF AZ(2)>0 THEN 820
770 A1Z=CZ-AZ(1):A3Z=AZ(1)-1
780 IF LPZ>A1Z THEN 800
790 P1Z=A1Z-LPZ:DZ=AZ(1)+LPZ-1:GOTO 890
800 IF LPZ>A3Z THEN 1210
810 P1Z=A3Z-LPZ:DZ=LPZ-1:GOTO 890
820 A1Z=CZ-AZ(1):A2Z=AZ(1)-AZ(2)-1:A3Z=A
Z(2)-1
830 IF LPZ>A1Z THEN 850
840 P1Z=A1Z-LPZ:DZ=AZ(1)+LPZ-1:GOTO 890
850 IF LPZ>A2Z THEN 870
860 P1Z=A2Z-LPZ:DZ=AZ(2)+LPZ-1:GOTO 890
870 IF LPZ>A3Z THEN 1210
880 P1Z=A3Z-LPZ:DZ=LPZ-1
890 POZ=INT(RND(1)*(P1Z+1))+DZ:C1Z(I)=PO
Z+1
900 FOR N=1 TO LPZ
910 A$=MID$(P$(I),N,1):M$(ROZ,POZ+2-N)="[
COMM6]" + A$
920 NEXT N:MZ(I)=2
930 GOTO 1470
940 AZ(1)=0:AZ(2)=0:COZ=INT(RND(1)*CZ)+1
:LPZ=LEN(P$(I)):C1Z(I)=COZ:DZ=0
950 FOR J=RZ TO 1 STEP -1
960 Z$=M$(J,COZ)
970 IF Z$="" THEN 1010
980 DZ=DZ+1
990 IF DZ>2 THEN 670
1000 AZ(DZ)=J
1010 NEXT J
1020 IF AZ(1)=0 AND AZ(2)=0 THEN P1Z=RZ-
LPZ:DZ=LPZ-1:GOTO 1160
1030 IF AZ(2)>0 THEN 1090

```

## EJEMPLARES ATRASADOS DE COMMODORE WORLD

# Commodore

## WORLD

8	11	12	13	14	16
17	18	19	20	21	

Precio del ejemplar 315 ptas.

Forma de pago: sólo por cheque o giro.

(Núms. anteriores están agotados).

Peticionario

Calle

Población

Nº TELEF.

D. P. Provincia

☐ Incluye cheque por valor de ..... pesetas+75 de gastos de envío.☐ Envío giro nº ..... por ..... pesetas.





# OLABORACIONES

```

1040 A1%=R%-A2%(1):A3%=A2%(1)-1
1050 IF LP%>A1% THEN 1070
1060 P1%=A1%-LP%:D%=A2%(1)+LP%-1:GOTO 116
0
1070 IF LP%>A3% THEN 670
1080 P1%=A3%-LP%:D%=LP%-1:GOTO 1160
1090 A1%=R%-A2%(1):A2%=A2%(1)-A2%(2)-1:A3%=
A2%(2)-1
1100 IF LP%>A1% THEN 1120
1110 P1%=A1%-LP%:D%=A2%(1)+LP%-1:GOTO 116
0
1120 IF LP%>A2% THEN 1140
1130 P1%=A2%-LP%:D%=A2%(2)+LP%-1:GOTO 116
0
1140 IF LP%>A3% THEN 670
1150 P1%=A3%-LP%:D%=LP%-1
1160 PO%=INT(RND(1)*(P1%+1))+D%:R1%(I)=P
O%+1
1170 FOR N=1 TO LP%
1180 A$=MID$(P$(I),N,1):M$(PO%+2-N,CO%)=
"[COMM6]" + A$
1190 NEXT N:M%(I)=3
1200 GOTO 1470
1210 A2%(1)=0:A2%(2)=0:CO%=INT(RND(1)*C%)+
1:LP%=LEN(P$(I)):C1%(I)=CO%:D%=0
1220 FOR J=1 TO R%
1230 Z$=M$(J,CO%)
1240 IF Z$="" THEN 1280
1250 D%=D%+1
1260 IF D%>2 THEN 400
1270 A2%(D%)=J
1280 NEXT J
1290 IF A2%(1)=0 AND A2%(2)=0 THEN P1%=R%-
LP%:D%=0:GOTO 1430
1300 IF A2%(2)>0 THEN 1360
1310 A1%=A2%(1)-1:A3%=R%-A2%(1)
1320 IF LP%>A1% THEN 1340
1330 P1%=A1%-LP%:D%=0:GOTO 1430
1340 IF LP%>A3% THEN 400
1350 P1%=A3%-LP%:D%=A2%(1):GOTO 1430
1360 A1%=A2%(1)-1:A2%=A2%(2)-A2%(1)-1:A3%=R
%-A2%(2)
1370 IF LP%>A1% THEN 1390
1380 P1%=A1%-LP%:D%=0:GOTO 1430
1390 IF LP%>A2% THEN 1410
1400 P1%=A2%-LP%:D%=A2%(1):GOTO 1430
1410 IF LP%>A3% THEN 400
1420 P1%=A3%-LP%:D%=A2%(2)
1430 PO%=INT(RND(1)*(P1%+1))+D%:R1%(I)=P
O%+1
1440 FOR N=1 TO LP%

```

```

1450 A$=MID$(P$(I),N,1):M$(PO%+N,CO%)="[
COMM6]" + A$
1460 NEXT N:M%(I)=4
1470 NEXT I
1480 FOR I=1 TO R%
1490 FOR J=1 TO C%
1500 IF M$(I,J)="" THEN M$(I,J)=CHR$(INT
(RND(1)*26)+65)
1510 NEXT J
1520 NEXT I
1530 N$="123456789012345678901234567890"
:X$="[30SHIFTC]"
1540 PRINT:PRINT:PRINT "[2CRSRR]TIEMPO P
ARA BUSCAR CADA PALABRA"
1550 PRINT "[CRSRR][33COMM]"
1560 INPUT "[CRSRR]MINUTOS";R2%(0)
1570 IF R2%(0)<0 THEN 1560
1580 INPUT "[CRSRR]SEGUNDOS";C2%(0)
1590 IF C2%(0)<0 THEN 1580
1600 BS=3600*R2%(0)+60*C2%(0)
1610 PRINT "[CLR][CRSRR][3CRSRR][COMM8]"
:LEFT$(N$,C%):PRINT "[BLK][2CRSRR][SHIFT
U]";LEFT$(X$,C%);"[SHIFTI]"
1620 FOR I=1 TO R%
1630 PRINT "[COMM8][CRSRR]";CHR$(64+I);"
[BLK][SHIFTB]";
1640 FOR J=1 TO C%
1650 C$=M$(I,J):IF LEN(C$)=2 THEN C$=RIG
HT$(C$,1)
1660 PRINT C$;
1670 NEXT J
1680 PRINT "[SHIFTB]"
1690 NEXT I
1700 PRINT "[2CRSRR][SHIFTJ]";LEFT$(X$,C
%);"[SHIFTK]"
1710 FOR I=1 TO MW%
1720 PRINT "[HOM]";LP%=LEN(P$(I))
1730 FOR J=1 TO LP%
1740 PRINT TAB(35)" ";MID$(P$(I),J,1)
1750 NEXT J
1760 MA%=21-LP%
1770 FOR J=1 TO MA%:PRINT TAB(36)" ";NEX
T J
1780 PRINT TAB(34)"ROW:[CRSRR][4CRSRL]CO
L:"
1790 VOL 8:SOUND 1,600,15
1800 IT=TI
1810 IF TI-IT>BS THEN FI=BS+1:GOTO 1980
1820 GET A$:IF A$="" THEN 1810
1830 FT=TI
1840 PO%=ASC(A$)

```

**Commodore**  
**WORLD**

Barquillo, 21, 3º Izda.  
Teléf.: 231 23 88/95  
28004 MADRID

Sant Gervasi de Cassoles, 39-despacho 4  
Teléf.: 212 73 45 / 212 88 48  
08022 BARCELONA

```

1850 IF POZ<65 OR POZ>64+RZ THEN 1810
1860 FI=FI-IT
1870 PRINT "[2CRSRU]";TAB(38)A$
1880 GETKEY Z$
1890 POZ=ASC(Z$)
1900 IF POZ<48 OR POZ>57 THEN 1880
1910 PRINT TAB(38)Z$;" [CRSRL]";
1920 GETKEY S$
1930 POZ=ASC(S$)
1940 IF POZ<48 OR POZ>57 THEN 1920
1950 PRINT S$
1960 A$(0)=VAL(Z$+S$)
1970 IF R1Z(I)=POZ-64 AND C1Z(I)=A$(0) T
HEN P$(I)="[COMM6]" + P$(I);GOTO 2130
1980 FWZ=FWZ+1
1990 ON MZ(I) GOTO 2000,2030,2060,2090
2000 POZ=C1Z(I)-1
2010 FOR N=1 TO LPZ
2020 A$=MID$(P$(I),N,1);M$(R1Z(I),POZ+N)
="[COMM4]" + A$;NEXT N;GOTO 2120
2030 POZ=C1Z(I)-1
2040 FOR N=1 TO LPZ
2050 A$=MID$(P$(I),N,1);M$(R1Z(I),POZ+2-
N)="[COMM4]" + A$;NEXT N;GOTO 2120
2060 POZ=R1Z(I)-1
2070 FOR N=1 TO LPZ
2080 A$=MID$(P$(I),N,1);M$(POZ+2-N,C1Z(I
))="[COMM4]" + A$;NEXT N;GOTO 2120
2090 POZ=R1Z(I)-1
2100 FOR N=1 TO LPZ
2110 A$=MID$(P$(I),N,1);M$(POZ+N,C1Z(I))
="[COMM4]" + A$;NEXT N
2120 P$(I)="[COMM4]" + P$(I)
2130 FD=FD+FI
2140 NEXT I
2150 GS=INT(FD/60);MN=INT(GS/60);SG=GS-M
N*60
2160 HS=PEEK(148)+60*PEEK(149)+3600*PEEK
(150)
2170 PRINT "[CLR]";IF FD<HS AND FWZ=0 TH
EN HS=FD;PRINT "[CRSRD][2CRSRR]UEVO RECO
RD!!";PRINT;GOTO 2210
2180 IF HS=933555 THEN 2230
2190 PRINT "[CRSRD][2CRSRR]RECORD"
2200 PRINT "[CRSRR][8COMMV]";GOTO 2220
2210 POKE 150,MN;POKE 149,SG;POKE 148,HS
-3600*MN-60*SG
2220 PRINT "[2CRSRU]";TAB(27)INT(HS/3600
);"";INT(HS/60)-60*INT(HS/3600);""
2230 Z$="";S$="";IF MN<1 THEN Z$="S"
2240 IF SG<1 THEN S$="S"
2250 PRINT "[2CRSRD][2CRSRR]TIEMPO TARDA

```

```

DO"
2260 PRINT "[CRSRR][16COMMV]"
2270 PRINT "[CRSRD]";MN;"MINUTO"+Z$
2280 PRINT SG;"SEGUNDO"+S$;PRINT;PRINT
2290 PRINT "[2CRSRR]PALABRAS ENCONTRADAS
";TAB(33)NMZ-FWZ
2300 PRINT "[CRSRR][22COMMV]";PRINT;PRIN
T
2310 PRINT "[2CRSRR]PALABRAS NO ENCONTRA
DAS";TAB(33)FWZ
2320 PRINT "[CRSRR][25COMMV]"
2330 PRINT;PRINT;PRINT TAB(12)"PULSA UNA
TECLA";GETKEY A$
2340 PRINT "[CLR][CRSRD]";TAB(4)"PALABRA
S";TAB(25)"POSICION"
2350 PRINT TAB(3)"[10COMMV]";TAB(24)"[10
COMMV]"
2360 A$(0)="HORIZONTAL";A$(1)="H. INVERSO
";A$(2)="VERTICAL";A$(3)="V. INVERSO"
2370 FOR N=1 TO NMZ
2380 PRINT "[CRSRR]";P$(N);"[BLK]";TAB(2
0)CHR$(64+R1Z(N));" -";C1Z(N);TAB(29)A$(
MZ(N)-1)
2390 NEXT N
2400 PRINT;PRINT "[CRSRR]OTRA PARTIDA S/
N? O MIRAS LA SOLUCION?[CRSRU]";GETKEY A
$
2410 IF A$="S" THEN 60
2420 IF A$="N" THEN PRINT "[CLR]";END
2430 PRINT "[CLR][CRSRD][5CRSRR][COMM8]"
;LEFT$(N$,CZ);PRINT "[BLK][4CRSRR][SHIFT
U]";LEFT$(X$,CZ);"[SHIFTI]"
2440 FOR I=1 TO RZ
2450 PRINT "[COMM8][3CRSRR]";CHR$(64+I);
"[BLK][SHIFTB]";
2460 FOR J=1 TO CZ
2470 PRINT "[BLK]";M$(I,J);
2480 NEXT J
2490 PRINT "[BLK][SHIFTB]"
2500 NEXT I
2510 PRINT "[4CRSRR][SHIFTJ]";LEFT$(X$,C
Z);"[SHIFTK]"
2520 GETKEY A$;GOTO 2340
3000 DATA ORDENADOR,CASSETTE,TECLADO,COM
ECTOR,MEMORIA,COMMODORE,ORGANIGRAMA
3010 DATA TELEVISION,PROGRAMA,MONITOR,VA
RIABLE,IMPRESORA,PANTALLA,DIMENSION,GRAF
ICO
3020 DATA JOYSTICK,PLOTTER,PADDLE,BASIC,
SISTEMA,DIAGRAMA,CONFIGURACION,RESOLUCIO
N,*

```

## SERVICIO DE CINTAS

De programas aparecidos en Commodore World.

Titulo del programa ..... publicado en nº ☐  
 Titulo del programa ..... publicado en nº ☐  
 Titulo del programa ..... publicado en nº ☐

Precio por cinta 995 pesetas. Gastos de envío 75 pesetas. Forma de pago: sólo por cheque o giro.

Peticionario .....  
 Calle ..... Nº ..... TELEF. ....

Población ..... D.P. .... Provincia .....

☐ Incluyo cheque por ..... pesetas.  
☐ Envío giro nº ..... por ..... pesetas.

Programa para VIC-20 ☐ Programa para  
 Programa para C-64 ☐ C-128 ☐

Si se desea disco, acogerse al servicio Commodore World en disco con todos los programas del nº correspondiente.



## Cuentas corrientes

C-64, C-128 (modo 64)

Antonio Castellanos de Toro  
C/ Mínimas, 2  
Daimiel (Ciudad Real)

Algunas veces nos encontramos con verdaderos problemas a la hora de tener que sumar los debes o

haber de un libro de cuentas corrientes. Con este programa podemos llevar un libro de este tipo paralelamente con el ordenador y de esta forma cometer menos errores, comprobar de vez en cuando y obtener del programa la suma de cantidades.

Sólo lleva un menú principal y por él se acceden a todas las posibilidades.

El menú es el siguiente:

1) Operaciones. Por esta opción, el usuario introduce los datos y si es la primera vez, también el nombre del banco. Si las respuestas son mayores que los caracteres permitidos (fecha=4, concepto=6, debe=6 y haber=6), el ordenador no dejará pasar la respuesta; por lo tanto, no nos preocupemos de estar contando las letras.

2) Listado. Por esta opción obtenemos un listado de todo lo almacenado por el medio anterior en un formato agradable y al final un total de debe, haber y saldo. El total del saldo es la resta del debe y haber y por lo tanto si no hay errores debe coincidir con el saldo final. Cuando terminemos con una página, el ordenador esperará a que pulsemos una tecla para pasar a otra. Si no hay más, volverá al menú principal. Si tenemos varias páginas completas y no queremos seguir, basta con pulsar la tecla F7 y volveremos al menú principal. Después de cada introducción, es conveniente hacer un listado de seguridad para cerciorarnos de que la entrada ha sido la correcta.

3) Modificaciones. Por esta opción sólo podremos alterar el contenido de la última entrada por lo que es bueno hacer un listado de seguridad para cerciorarnos antes de seguir adelante. Si es la primera vez, también nos pedirá el nuevo nombre del banco.

4) Cargar datos. Por esta opción podremos cargar de cinta todos los datos que anteriormente hayan sido grabados. Tenemos que introducir el número de fichero con que los datos han sido grabados.

5) Salvar datos. Por esta opción podemos grabar en cinta todo lo que tengamos en memoria. Debemos introducir un número de fichero. Puede ser el uno. Si cargamos datos y después queremos grabarlos otra vez con modificaciones o sin ellas, hay que poner en el fichero un número diferente a con el que han sido grabados.

6) Salir del programa. Como su nombre indica sirve para salir del programa y por lo tanto terminar la ejecución. El ordenador hará un comentario para recordarnos que debemos grabar los datos en cinta y tendremos la posibilidad de volver al menú principal.

7) Balance de movimientos. Por esta opción el ordenador nos pro-

porcionará una especie de estadística en barras horizontales que, aunque no muy exactamente, nos dejará ver la evolución de todos los saldos a lo largo de las cuentas corrientes. Al final también nos mostrará el saldo medio de todos nuestros movimientos.

Esto es todo; el programa está preparado para 200 registros pero supongo que funcionará con más. Espero que os animéis a enviar programas útiles y no tantos juegos, si nos os volvéis locos después de teclearlo.



```

10 REM ***** .60
20 REM *** CUENTAS CORRIENTES *** .162
30 REM *** POR *** .242
40 REM *** ANTONIO CASTELLANOS*** .216
50 REM *** 19-XII-1985 *** .198
60 REM ***** .110
70 POKE 53280,0:POKE 53281,0:PRINT"
[VEL]"
100 DIMA$(200,1):DIMA(200,2) .224
105 A=0:X=1:H=0:PRINT"[CLR]":PRINTT .211
AB(6);"[COMM1][20SHIFT*][COMMS]":U1
=0:U2=0
110 PRINTTAB(6);"[CSHIFT-J]CUENTAS[3S .240
PCJCORRIENTES[SHIFT-J]"
115 PRINTTAB(6);"[COMM2][20SHIFT*][ .231
COMMJ]"
120 PRINT:PRINT:PRINT .248
130 PRINT"[7SPC]1.- OPERACIONES.":P .208
RINT
140 PRINT"[7SPC]2.- LISTADOS.":PRIN .172
T
150 PRINT"[7SPC]3.- MODIFICACIONES. .130

```

# Commodore

# WORLD

Barquillo, 21, 3º Izda.  
Teléf.: 231 23 88/95  
28004 MADRID

Sant Gervasi de Cassoles, 39-despacho 4  
Teléf.: 212 73 45 / 212 88 48  
08022 BARCELONA

```

":PRINT
160 PRINT"[7SPC]4.- CARGAR DATOS.":.86
PRINT
170 PRINT"[7SPC]5.- SALVAR DATOS.":.160
PRINT
175 PRINT"[7SPC]6.- SALIR DEL PROGR .111
AMA.":PRINT
177 PRINT"[7SPC]7.- BALANCE DE MOVI .147
MIENTOS.":PRINT
180 PRINT"[7SPC]MOVIMIENTOS.":T:PR .54
INT:PRINT
190 INPUT"[2SPC]INTRODUZCA LA OPCIO .106
N DESEADA.":L:PRINT"[CLR]"
200 ONLGOTO1000,2000,3000,4000,5000 .12
,6000,7000
1000 PRINT"[HOM][7SPC][RVSON][2SPC] .225
ENTRADA DE DATOS[2SPC][RVSOFF]":PRI
NT:PRINT
1010 X=1:T=T+1 .39
1016 PRINT:PRINT:PRINT .123
1020 IFT=1THENINPUT" - NOMBRE DEL B .139
ANCO.":G$:IFLEN(G$)>22THENGOTO1020
1040 PRINT:INPUT" - CONCEPTOS.":A$( .150
T,0)
1041 IFLEN(A$(T,0))>6THENPRINT"[CRS .61
RU][27SPC][2CRSRU]":GOTO1040
1050 PRINT:INPUT" - FECHA (DDMM.":.158
A$(T,1)
1051 IF LEN(A$(T,1))>4 THEN PRINT"[ .9
CRSRU][25SPC][2CRSRU]":GOTO1050
1060 PRINT:INPUT" - DEBE O HABER (D .102
/H.":P$:PRINT
1070 IFP$="H"THEN INPUT" - IMPORTE .204
DEL INGRESO.":A(T,1):A(T,0)=0
1071 IF LEN(STR$(A(T,1)))>7 THEN PR .231
INT"[CRSRU][29SPC][CRSRU]":GOTO1070
1080 IFP$="D"THEN INPUT" - IMPORTE .132
DEL GASTO.":A(T,0):A(T,1)=0
1081 IF LEN(STR$(A(T,0)))>7 THEN PR .19
INT"[CRSRU][28SPC][CRSRU]":GOTO1080
1085 IFP$<>"H"ANDP$<>"D"THENPRINT .143
"[2CRSRU][29SPC][2CRSRU]":GOTO1060
1300 GOTO105
2000 PRINT"[HOM][COMMJ][4SHIFT*][CO .53
MMS]"
2010 PRINT"[SHIFT-][4SPC][COMMJ][6S .109
HIFT*][COMMJ][7SHIFT*][COMMJ][7SHIF
T*][COMMJ][9SHIFT*][COMMJ]"
2014 PRINT"[SHIFT-][4SPC][SHIFT-][6 .247
SPC][SHIFT-][7SPC][SHIFT-][7SPC][SH
IFT-][9SPC][SHIFT-]"

```

```

2015 PRINT"[COMMJ][4SHIFT*][SHIFT+] .108
[6SHIFT*][SHIFT+][7SHIFT*][SHIFT+][
7SHIFT*][SHIFT+][9SHIFT*][COMMJ]"
2020 FORB=1TO16 .77
2030 PRINT"[SHIFT-][4SPC][SHIFT-][6 .7
SPC][SHIFT-][7SPC][SHIFT-][7SPC][SH
IFT-][9SPC][SHIFT-]"
2040 NEXTB .27
2050 PRINT"[COMMJ][4SHIFT*][COMMJ][ .6
6SHIFT*][COMMJ][7SHIFT*][COMMJ][7SH
IFT*][COMMJ][9SHIFT*][COMMJ]"
2060 PRINT"[HOM][6CRSRU]":G$:PRINT" .164
[CRSRU]FECH[CRSRU][25CRSRU]HOJA.":H
, "[CRSRU]"
2065 PRINT"[CRSRU]1985[CRSRU]CONCEP .97
[3CRSRU]DEBE[3CRSRU]HABER[4CRSRU]SA
LDO[CRSRU]":PRINT
2070 FORN=XTO T .34
2071 IFN=1THENA(N,2)=A(N,1)-A(N,0): .237
GOTO2080
2075 A(N,2)=A(N-1,2)+A(N,1)-A(N,0) .89
)
2080 Q1=LEN(A$(N,1)):Q2=LEN(A$(N,0) .94
):Q3=LEN(STR$(A(N,0))):Q4=LEN(STR$(
A(N,1)))
2090 Q5=LEN(STR$(A(N,2))) .254
2100 PRINT"[CRSRU]"SPC(4-Q1)A$(N,1) .86
"[CRSRU]"SPC(6-Q2)A$(N,0)"[CRSRU]"S
PC(7-Q3)A(N,0)SPC(7-Q4)A(N,1):
2110 PRINTSPC(9-Q5):A(N,2) .76
2111 U1=U1+A(N,0):U2=U2+A(N,1) .229

2120 IFN=X+15THEN H=H+1:GOTO2130 .60
2122 NEXTN .206
2123 P=P+1 .63
2130 PRINT"[HOM][20CRSRU]" .72
2140 PRINT"TOTAL DEBE.":U1 .160
2150 PRINT"TOTAL HABER.":U2 .130
2160 PRINT"TOTAL SALDO.":U2-U1 .40
2170 GETS$:IFS$=""THEN2170 .180
2175 IFS$="[F7]" THEN GOTO105 .127
2180 IFP=1THENP=0:H=0:GOTO105 .198
2600 X=X+16 .110
2700 PRINT"[CLR]":GOTO 2000 .178
3000 PRINT"[9SPC][RVSON][2SPC]MODIF .15
ICACIONES[2SPC][RVSOFF]"
3010 PRINT:PRINT .159
3020 PRINT"[4SPC]POR ESTA OPCION SO .137
LO PUEDE ALTERAR"
3030 PRINT"[2SPC]EL CONTENIDO DEL U .149
LTIMO MOVIMIENTO."

```

## EJEMPLARES ATRASADOS DE «CLUB COMMODORE»

Primera época (septiembre-1982 - enero-1984)

Para poder satisfacer la creciente demanda de Club Commodore, agotada en todos sus números, hemos puesto en marcha un Servicio para suministrar fotocopias de los ejemplares que nos sean solicitados.

SERVICIO DE FOTOCOPIAS - NUMERO DE LA EDICION SOLICITADA

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

Peticionario. D

Calle

Poblacion

Nº

TELEF.

D P

Provincia

Forma de pago sólo por cheque

Precio de la edición fotocopiada: 295 ptas.

La colección completa del 0 al 15: 2.950 ptas. + 150 ptas. por gastos de envío.

Incluyo cheque por ..... ptas.

Envío giro nº

por

pesetas.





## OLABORACIONES

```

3040 PRINT:PRINT:U1=U1-A(T,0):U2=U2 .87
-A(T,1)
3045 IFT=1 THEN INPUT"[2SPC]- BANCO .78
:":G$
3050 INPUT"[2SPC]- FECHA REAL <DDMM .77
>":A$(T,1)
3060 INPUT"[2SPC]- CONCEPTO REAL:": .207
A$(T,0)
3070 INPUT"[2SPC]- DEBE O HABER <D/ .127
H>":P5$
3080 IF P5$="D" THEN INPUT"[2SPC]- .208
GASTO REAL:":A(T,0):A(T,1)=0:U1=U1+
A(T,0)
3090 IF P5$="H" THEN INPUT"[2SPC]- .140
INGRESO REAL:":A(T,1):A(T,0)=0:U2=U
2+A(T,1)
3100 IF P5$<>"D" AND P5$<>"H" THEN P .6
RINT"[2CRSRU][24SPC]":GOTO3070
3110 GOTO 105 .140
4000 INPUT"NUMERO DE FICHERO:":L1:0 .57
PENL1,1,0,"CUENT.CORRIENT"
4010 INPUT#L1,T:INPUT#L1,G$ .53
4020 FORB=1TOT .129
4030 INPUT#L1,A$(B,0):INPUT#L1,A$(B .135
,1)
4050 INPUT#L1,A(B,0):INPUT#L1,A(B,1 .105
*)
4055 NEXTB .2

```

**Commodore**

**WORLD**

**¡¡SUSCRIBETE!!**

Boletín encartado

```

4060 GOTO105 .69
5000 INPUT"NUMERO DE FICHERO:":L1:0 .105
PENL1,1,1,"CUENT.CORRIENT."
5010 PRINT#L1,T:PRINT#L1,G$ .133
5020 FORG=1 TO T .149
5030 PRINT#L1,A$(G,0):PRINT#L1,A$(G .113
,1)
5040 PRINT#L1,A(G,0):PRINT#L1,A(G,1 .77
)
5050 NEXTG .17
5060 CLOSE1 .207
5070 GOTO 105 .59
6000 PRINT" [RVSON][2SPC]NO OLVIDE .41
GRABAR DATOS EN CINTA[2SPC][RVSOFF]
"
6010 PRINT:PRINT:PRINT .17
6020 PRINT"[4SPC]PULSE S=VOLVER AL .13
MENU F=FINAL"
6030 GET A$:IF A$="" THEN 6030 .123
6040 IF A$="S" THEN GOTO 105 .125
6050 IF A$="F" THEN 6070 .123
6060 IF A$<>"F" AND A$<>"S" THEN 60 .23
30
6070 END .207
7000 REM CONVERSION DE NUMEROS EN G .77
RAFIOS
7001 PRINT"[8SPC]BALANCE DE MOVIMIE .178
NTOS[9SPC]":PRINT:PRINT
7002 C=10:VV=0 .105
7010 FORY=1TOT .243
7020 IF A(Y,2)>VV THENVV=A(Y,2) .5
7030 NEXTY .101
7050 FORZ=1 TO T .35
7051 PRINTA$(Z,0),A$(Z,1) .190
7055 HH=(38*A(Z,2))/VV .98
7060 FORFF=0 TO HH .215
7070 PRINT"[RVSON] "; .125
7080 NEXTFF:PRINT .17
7090 IF Z=C THEN 7120 .61
7100 NEXTZ .179
7102 FORQ=1TOT:Y=Y+A(Q,2):NEXT .65
7103 PRINT:PRINT:PRINT:PRINT"[4SPC] .32
SALDO MEDIO:":Y/T
7104 PRINT:PRINT"[6SPC]UNA TECLA PA .189
RA EL MENU[7SPC]"
7105 GET A$:IF A$="" THEN 7105 .150
7110 GOTO 105 .59
7120 GET A$:IF A$="" THEN 7120 .161
7125 IF A$="[F7]" THEN GOTO 105 .88
7130 C=C+10:PRINT"[CLR]":GOTO 7100 .211

```

**Commodore**

**WORLD**

Barquillo, 21, 3º Izda.  
Teléf.: 231 23 88/95  
28004 MADRID

Sant Gervasi de Cassoles, 39-despacho 4  
Teléf.: 212 73 45 / 212 88 48  
08022 BARCELONA

# DIRECTORIO

## casa de software s.a.

c/ aragón, 272, 8º, 6.ª  
tel. 215 69 52 - 08007 barcelona

- Software profesional para C-64
- Con distribución productos DIGITAL RESEARCH

## RADIO WATT

- ORDENADORES PERSONALES
- ACCESORIOS INFORMATICA
- COMPONENTES ELECTRONICOS
- TELECOMUNICACIONES

Paseo de Gracia 126-130  
Tel. 237 11 82\*. 08008 BARCELONA

### EN HUELVA

Commodore  
Spectrum  
Nixdorf

### INFORMATICA COMPUTER LOG

C/ Puerto, 6  
HUELVA. Teléf.: 25 81 99

## IEEE 58 MICROTEISA

c/ Miguel Yuste, 16  
Telf. 204 51 98 - Madrid

## COMMODORE

ORDENADORES-PERIFERICOS-SOFTWARE  
SERVICIO TECNICO

### PARA COMMODORE 64

Convierte tu ordenador inglés  
en un ordenador español  
mediante este cartucho.  
Solamente £ 75 (libras esterlinas)  
incluyendo envío aéreo.

Enviar pedido a:

Premilink Exports - 5, Fairholme Gardens  
London N. 3 - T: 01-346 1044

### COMPARE LOS PRECIOS

COMMODORE 64	54.500
DATASETTE	10.000
FAST-TURBO MENUE	8.500
SOLO FLIGHT	3.800
COLOSSUS CHESS 2.0	2.800
ZAXXON	2.300
<b>TOTAL</b>	<b>81.900</b>

**POR SOLO  
59.900 Ptas.**

Commodore 16.....	23.900 pts.
Joystick Quickshot II .....	2.450 pts.
Discos RPS 1D (caja de plástico)....	340 pts.
Discos RPS 2D (caja de cartón).....	395 pts.
Monitor Ciagei con sonido .....	22.900 pts.

Oferta de Commodore 128

Consulta precios

### ASTOC DATA

Apartado de Correos, 695  
SANTIAGO DE COMPOSTELA  
Tel. (981) 59 95 33

## MICROWORLD

### HACEMOS FACIL LA INFORMATICA

- SINCLAIR • SPECTRAVIDEO
- COMMODORE • DRAGON
- AMSTRAD • APPLE
- SPERRY UNIVAC

Modesto Lafuente, 63  
Telf. 253 94 54  
28003 MADRID

José Ortega y Gasset, 21  
Telf. 411 28 50  
28006 MADRID

Fuencarral, 100  
Telf. 221 23 62  
28004 MADRID

Ezequiel González, 28  
Telf. 43 68 65  
40002 SEGOVIA

Colombia, 39-41  
Telf. 458 61 71  
28016 MADRID

Padre Damián, 18  
Telf. 259 86 13  
28036 MADRID

Avda. Gaudi, 15  
Telf. 256 19 14  
08015 BARCELONA

Stuart, 7  
Telf. 891 70 36  
ARANJUEZ (Madrid)

## CIGUS & INFORMATICA

COMMODORE 16, 64, 128  
COMMODORE PC  
PERIFERICOS  
SOFTWARE  
HARDWARE

Horta Novella, 128 Tel. 725 85 68 (SABADELL)

Pide nuestro  
CATÁLOGO



## VENTAMATIC

• VEN A CONOCERNOS. Somos los SUPER-ESPECIALISTAS del COMMODORE 64 y lo tenemos TODO para tu COMMODORE 64 (incluyendo lo último en accesorios y programas musicales y MIDI). SOLICITA CATALOGO COMPLETO.

• VENTAMATIC - c. Corcega, 89, entlo 08029 BARCELONA - Tel. (93) 230 97 90. Metro EN-TENZA Línea V Bus 41, 27, 15, 54, 66.

Departamento de Envios y Venta por Correo:  
• VENTAMATIC - Avda. de Rhode, 253 - ROSES (Girona) - Tel. (972) 25 79 20



A PUBLICATION OF  
CW COMMUNICATIONS

### SOFTWARE ESPAÑA

Avda. de Arteijo, 19  
14004 La Coruña  
Teléf. 25 51 72

Especializados  
en software para  
Commodore-64  
Spectrum  
y MSX

## MICROS GARDEN SA

- ORDENADORES PERSONALES -

- Sinclair Spectrum - Plus - QL
- Spectravideo 328 y MSX
- Atari 800 XL y 600 XL
- Dragon 32 y 64
- Commodore 64
- Oric Atmos
- Amstrad
- Epson

— PERIFERICOS Y ACCESORIOS  
— SOFTWARE PARA TODAS LAS MARCAS  
— CURSOS DE BASIC  
— GRAN SURTIDO EN LIBROS Y REVISTAS

Francisco Silvela, 19  
Teléfono 401 07 27. 28028 MADRID



C

## ARTA BLANCA...

## EL "AUDIO" SOLUCIONADO

Me dirijo a vosotros para intentar ayudar a responder a la pregunta que efectuó un lector en el número 21, de diciembre, de vuestra revista. La pregunta era: "¿Cuál es la función del Input de audio en el conector de audio/vídeo y cómo y de qué forma se puede usar?"

Yo, por si sirve de ayuda, puedo deciros que he experimentado algo sobre ese tema.

Si se le conecta al C-64 una entrada de sonido oiremos dicho sonido por el altavoz de nuestro monitor, para eso dispone el ordenador de un mezclador de audio interno, con el que podemos mezclar la señal introducida con la señal proveniente del chip SID.

Asimismo disponemos también de unas posiciones de memoria a disposición del tratamiento de dicha señal:

54295 - control de resonancia de

filtro/control de voz de entrada.

BIT 3 - filtro de entrada externa.

Con lo que podemos filtrar la señal de entrada como si de otra voz se tratara, podemos decir que nuestro C-64 posee 4 voces y a su vez la cuarta pueden ser cientos.

Advertencia: ojo a la supervisión de los cables, ya que aparte de señales de audio disponemos en el conector de señales de video.

Otra cosa, he teclado el programa Turbosave del número de noviembre y lo veo interesante, pero he observado que es el único turbo con el que puedo cargar programas sin desconectar el cable de mi TV. ¿Son tan distintos los comerciales o son problemas de grabación de las cintas originales?

Antonio Francisco Saá Quintas  
c/ El Almendro, 120  
Aguadulce (ALMERIA)

*Puede que tengas razón en que los turbos comerciales a veces van mal grabados, no es que sea problema del turbo que utilicen, porque casi todos trabajan de la misma forma y aproximadamente a la misma velocidad, sino que la forma de hacer copias (a nivel industrial) no es del todo ortodoxa. En vez de utilizar un ordenador para hacer cada cinta —que sería lo recomendable— se utilizan las mismas máquinas que para reproducir las cassettes musicales. En una cassette musical no se nota apenas ruido, ni fallos de grabación, pero un ordenador sí que nota la diferencia. Además, estas cintas se graban a doble velocidad y por las dos caras a la vez.*

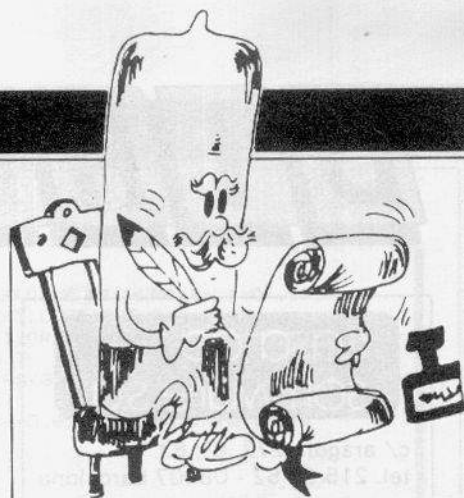
*Esto no quiere decir que todas las cintas comerciales sean malas, sino que algunas dan problemas. También es probable que el problema resida en tu azimuth, pero si consigues leer los programas medianamente bien no es recomendable ajustarlo.*

## NO HAY TANTOS FALLOS

Estoy suscrito a vuestra revista desde sus inicios, y la encuentro muy interesante. Poseo un C-64 y unos 200 programas, unos comprados y otros cambiados.

En el número 21, de diciembre, encontré en la revista el programa que necesitaba para tener en orden mis discos. Sin pensarlo di comienzo a teclearlo, tras unas horas de trabajo con ilusión pensando que al fin se acabarían mis problemas para buscar programas en mis discos y una vez terminado de teclear, hice RUN y, ¡sorpresa! salían patatas por todas partes.

Con paciencia lo verifiqué y



comencé a buscar errores, encontrando los siguientes:

— 15 H=0: IF CH THEN CH = 0 (es H y no CH).

— 19 tras H=0 :: A=0 (sólo una vez :).

— faltan comillas en las líneas 525-530-531-540-560-642

— en la 744 hay una L de más.

— en la línea 1006 y 1106 la última C\$ creo que es D\$.

— "syntax error" en la 132.

en fin y unos cuantos más.

Pero al final, al teclear la opción 5, el programa se queda bloqueado en: CALCULANDO/LISTA EN ORDENACION...

Mi comentario es el siguiente:

¿No es una pena que haya tantos errores en los programas? Teniendo en cuenta que los que compran estas revistas, en principio, no son programadores y si aficionados con ganas de aprender. ¿No dejarían de comprar esta revista desilusionados y buscar otra?

Mi pregunta es: ¿Habría posibilidad de obtener una fotocopia del listado original del programa "discos ordenados"? Si así fuera les agradecería que me lo enviaran, o una cinta con el programa.

Ricardo J. Noguera  
C./ Maestro Chapí, 4-1.º  
29002 MALAGA

En primer lugar veamos los "errores" que has encontrado:

Línea 15: esa línea está bien, equivale a IF CH <> 0 THEN...

Línea 19: dos veces dos puntos no provocan ningún error, el ordenador sigue ejecutando el programa como si nada.

Líneas de las comillas: cerrar comillas después de un PRINT o de cualquier otra cosa es opcional si a continuación no hay más sentencias. Además, no poniéndolas ahorras memoria.

Líneas 1006 y 1106: ahí tienes razón.

Línea 132: Esa línea nunca en la vida puede dar un syntax error. Poner los "," entre literales o variables en una sentencia PRINT también es opcional (a menos que se puedan confundir los nombres de las variables —dos variables numéricas seguidas— o el punto y coma sea el final de la línea).

M

## MEJORANDO LO PRESENTE

## VOYAGER MEJORADO

Las líneas que os mando a continuación corresponden al programa Voyager, para el Vic-20, publicado en el número 10 (diciembre 1985). Los números de línea sueltos indican que se han de suprimir del programa original. La mejora corresponde a la utilización del joystick, y se han de cambiar en el segundo listado.

620 01=RJOY(0)

630 IF01=1THENY1=Y1-1

640 IF01=2THENY1=Y1+1

Juan Bardera Bosch  
C/ Dalmacio Creixell, núm. 2. 1. 2  
08032 BARCELONA

— No hay tantos errores en los programas, si se nos cuela alguno es porque —aparte de que somos humanos— no hemos comprobado todas las opciones del programa a conciencia, lo cual es bastante complicadillo (ej. el error que citas en la línea 1006: no podemos comprobar todos los errores de disco posibles para darnos cuenta de que el programa escribe mal el sector en el que se produce el error). Donde más errores hay (cuando los hay) es en los listados que no están sacados por impresora, como sucedía hace mucho con las "magias".

— Es inútil que nos pidas una fotocopia del listado original porque lo que tenéis en la revista es el listado original. Lo único que media entre el ordenador y lo que veis publicado es nuestro "listador" para obtener los códigos de control, y ese programa lleva dos años funcionando perfectamente.

De cualquier manera: A partir de ahora ya no habrá más errores (¡crucemos los dedos!) gracias al nuevo sistema de suma de control. Si aún tenéis problemas podéis llamarnos por teléfono o enviarnos una cinta con el programa

que habéis tecleado y no os funciona para que os grabemos la versión correcta.

### MUSICA Y CODIGO MAQUINA

A) ¿Cómo consiguen algunos programas comerciales que suene música mientras el programa está cargando?

B) ¿Por medio de qué tipo de interrupciones puedo conseguir más de 8 sprites a la vez?

C) ¿Dónde puedo encontrar una descripción detallada de todos los mnemónicos de c.m. del C-64?

D) ¿Cuál es la impresora adecuada para mí (14 años) que me permita hacer trabajos presentables con mi C-64 y que esté relativamente bien de precio?

E) ¿Podéis aconsejarme un/unos buenos, buenos programas para mi C-64 que me permitan componer buenas melodías (a 2 ó 3 voces) para utilizarlas posteriormente?

F) Una petición: Algunos de mis compañeros de clase disponen de un, ejem, ejem... Spectrum 48K. Ellos erre que erre con que el Spectrum es un

ordenador de mejores prestaciones que el C-64, y yo que no, que no tiene sprites, que la resolución del C-64 es mejor, etc., pero ellos ni caso. Por favor, ¿podríais decir aquí cuál de los dos tiene mejores prestaciones?

Héctor Blanquer Tur  
c/ Marina, 368, ático 2.ª  
08025 BARCELONA

A) Los programas comerciales, entre ellos todos los que llevan el sistema de carga Novaload, utilizan una mezcla de sistema turbo-interrupciones que hacen que sea posible que suene música mientras se está cargando el programa. La música se va leyendo también durante la carga del programa. Este y otros sistemas de carga son muy complicados de utilizar y quedan reservados para los "profesionales".

B) Mediante las llamadas interrupciones por "barrido", una forma de programación que requiere un profundo conocimiento sobre código máquina y el funcionamiento del chip VIC-II que lleva incorporado el ordenador. Básicamente consiste en "parar" al ordenador en una línea determinada de la pantalla, ejecutar

# ¡¡¡ASOMBROSO!!! THE FINAL CARTRIDGE®

Ahora un  
NUEVO SISTEMA OPERATIVO EXTERNO  
para su Commodore 64 o su Commodore 128 (en modo 64)  
INCORPORA

- El Acceso a disco 5 veces más rápido (divide por 5 tanto el tiempo de carga como el de grabación de su programas).
- Un turbo-casete 10 veces más rápido para todas las operaciones (programas y ficheros) utilizando los comandos normales de su C64. Compatible con "TURBO TAPE 64".
- 24 K RAM más para la programación Basic.
- Numerosas instrucciones del Basic 4.0 (Catalog, Dload, Dsave, Dappent, etc).
- Nuevos comandos de ayuda a la programación como: Find, Auto, Renum (incluso Goto y Gosub), Del, Old, Help, etc.
- Un potente y sencillo interface para impresora tipo centronics (paralelo), que le permitirá aprovechar las posibilidades de su impresora, pero también reproducir los caracteres gráficos y los códigos de control del Commodore.
- Un fantástico monitor de código máquina, que podrá llamar en cualquier momento (si, en cualquier momento). Incluye todas las funciones habituales más otras excepcionales, como Bank Switching. Este monitor no ocupa memoria alguna y permite acceder fácilmente a cualquier posición de memoria, incluso debajo del Kernal.
- Vocado de pantalla (Hard Copy), en alta y baja resolución. Compatible con Koala Pad y programas similares.
- Varios comandos especiales como por ejemplo "Type" que le permitirá usar su ordenador y su impresora como una máquina de escribir electrónica.
- Las teclas de función preprogramadas: F3 = RUN, F7 carga y lista el directorio del disco, conservando el programa en memoria, etc.
- Un botón de Reset que permite Resetear todos los programas.
- Un interruptor on/off (este por si acaso... supongamos que no lo querrá usar).
- Extenso manual en castellano.

Todo esto en un cartucho totalmente transparente  
NO REQUIERE NINGUNA INSTALACION ESPECIAL  
(Enchufe el cartucho y no lo quite más)

Pídalo en su tienda o directamente por correo a HISPASOFT, S.A.

BOLETIN DE PEDIDO  
Ptas.



DISTRIBUIDOR EXCLUSIVO

HISPASOFT, S.A.

C/Coso, 87 · 6ºA · Tfno. (976) 399961 - 500001 · ZARAGOZA

— The Final Cartridge	×	13.900
— Cable Centronics	×	3.450
<input type="checkbox"/> Contra reembolso		<input type="checkbox"/> Talón adjunto
Nombre .....		
Dirección .....		
Población o Provincia .....		





nuestra rutina en código máquina, y seguir con el barrido. Si se habían definido sprites en la parte superior de la pantalla y en nuestra rutina los colocamos en la parte baja —por debajo de la línea en la que se ha producido la interrupción— el ordenador seguirá con el barrido de la pantalla haciendo aparecer los sprites (que son los mismos) en la parte inferior. Estamos preparando un artículo sobre el tema para aclarar el funcionamiento de este tipo de interrupciones que ofrecen un amplio campo de posibilidades en la programación.

C) En el "cursillo de código máquina" de la biblioteca Commodore World —es el número 1— aparece una lista de todos los mnemónicos del 6502, que son los mismos del 6510 del C-64. En la guía de referencia también aparecen todos con una explicación de su funcionamiento.

D) Si te compras una MPS-801 de Commodore vas a tener problemas, ya que es sólo de tracción y la calidad de letra deja mucho que desear. Las dos opciones más interesantes del mercado por la relación precio/prestaciones son la RITEMAN C+ y la SG-10. De estas dos impresoras hay varios modelos. Elegir el que más te convenga es tarea tuya, lo único que te podemos recomendar es que las veas funcionando para poder elegir mejor.

E) En el número 14 de nuestra revista tienes el programa SID-SERENADE, que utiliza la programación de interrupciones para generar los sonidos, con el que se pueden generar fácilmente canciones en varias voces. En ese mismo número, que es un especial sonido y música, hay reseñas de programas musicales para todos los gustos.

F) Lo único en lo que el Spectrum (ya sea 16K, 48K, o +) supera al C-64 es en el microprocesador Z-80. Pero sucede que al pobre Z-80 sólo lo utilizan... para la programación de juegos. En todo lo demás el C-64 barre el mapa al Spectrum (esto es una apreciación personal, por supuesto). Ni en velocidad y sencillez del Basic (con o sin ampliaciones), ni en cuanto a los programas disponibles, teclado, periféricos, etc., se pueden comparar. El precio es tema aparte. Un ordenador que en vez de teclas parece que tiene gomas de borrar y ni siquiera tiene interruptor de encendido tiene que costar menos a la fuerza. Perdón si alguien se siente ofendido, pero las cosas son como son...

## HAY QUE MEJORAR

Soy usuario del nuevo C-128 y espero que veamos aparecer pronto

programas y artículos sobre él en su revista.

Estoy suscrito a Commodore World desde su aparición y debo decirles que empiezo a sentirme algo defraudado. La revista se está convirtiendo en una repetición de los mismos programas de siempre (bingo, tres en raya, etc.). Yo creo que los usuarios de ordenador hemos pasado ya esa fase y lo que queremos ver en su revista son programas que nos descubran algo más del ordenador que unos cuantos juegos.

Echo en falta artículos interesantes sobre el Sistema Operativo, tanto del ordenador como del disco, mapas de memoria comentados, etc., creo que hay suficiente material para salir de la monotonía en que nos tenéis sumidos. Como ejemplo no hay más que leer vuestra colega americana RUN que por un precio parecido ofrece una información mucho más interesante y completa.

José María Martínez  
suscriptor 817

En parte tienes razón: algunos programas del mismo tema han salido repetidos, aunque siempre son programas que se van mejorando con el tiempo. Tienes que tener en cuenta que hay gente que no nos sigue desde el primer número y no conocen esos programas.

También tenéis un poco de culpa los lectores: las colaboraciones que nos enviáis (si las enviáis) son repeticiones hasta la saciedad de los mismos temas, por este orden: lotería primitiva (14 desde que existe), representación de funciones en alta resolución (en Basic por lo general), resolución de problemas matemáticos tan particulares que no sirven para nada en la "vida real",

pianos, ahorcados, froggers, monitores de c.m., quinielas, etc. Los que no pertenecen a esta "clasificación" son casi siempre juegos. Algunos meses es bastante difícil encontrar cuatro o cinco colaboraciones que merezcan la pena. ¡A ver si con esto del premio a la mejor colaboración os animáis un poquito! Por otro lado, estamos preparando artículos sobre esos temas que has citado: funcionamiento de la unidad de discos, sistema operativo, lenguajes, etc. Todo llegará. Este mes, como veréis, el artículo "RESET" no es de nadie de la revista, es de un amigo de Málaga que nos lo envió para su publicación. Estamos dispuestos a admitir este tipo de colaboraciones siempre y cuando tengan calidad suficiente para ser publicadas como artículos independientes. Enviadnos artículo completo (a máquina preferiblemente), cinta o disco con el programa/s, fotos o dibujos, etc., a la dirección de siempre indicando que se trata de un artículo, no de una colaboración. Lo estudiaremos y veremos si merece ser publicado (incluso puede haber dinerillo por medio).

## SKY TRAVEL... ¿CUANDO?

Soy un aficionado a la Astronomía y quisiera saber cuándo va a salir al mercado el programa de astronomía que se menciona en la revista del mes de diciembre, página 82 y si es posible también su precio y el distribuidor.

Pedro Crespo  
MADRID

Sky travel, que es el programa al que te refieres, no está en estos momentos disponible en España, igual que sucede con muchos programas que se están comercializando en el extranjero. Si alguien lo importara, en teoría podría ser Microelectrónica y Control, dado que ellos son distribuidores exclusivos de Commodore en España y Sky Travel es un producto de Commodore International. Mientras tanto puedes hacer lo que estamos haciendo muchos: comprarte revistas americanas, buscar en los anuncios la dirección del distribuidor y mandarles un "money order" o pagar con la tarjeta de crédito para conseguir el programa. Sky travel es uno de los mejores programas de astronomía que existen para el C-64.

Si nos quieres  
consultar cualquier duda  
que tengas, envíanos una  
carta a Commodore World  
C/ Barquillo, 21 - 3º Izqda.  
28004 MADRID

## EXCURSION POR LA MICROPROGRAMACION

# Rincón del Código Máquina-VIII

Por Diego ROMERO

**E**n el número 15 de Commodore World (mayo-1985) publicamos una rutina de código máquina que facilitaba la ordenación de matrices desde un programa Basic en el C-64, consiguiéndolo en un tiempo mucho menor que el empleado por rutinas de clasificación por orden alfabético en Basic.

Muchos lectores han reclamado una rutina de funcionamiento similar para el Vic-20 (la verdad es que hemos dejado un poco de lado al hermano pequeño de la gama Commodore) y, esperando que los usuarios de este "pequeño gigante" al que tantos buenos ratos le tenemos que agradecer, publicaremos entre este número y el próximo unas rutinas similares para el Vic.

En el número 15 publicamos la rutina de ordenación del C-64, y en el 16 la de búsqueda. En lugar de hacer esto mismo con el Vic, en este número incluimos las dos rutinas juntas.

La ventaja de velocidad que obteníamos en el C-64 mantiene la misma proporción en el caso del Vic-20, pero este último es aproximadamente un 10% más rápido que su hermano mayor. Esto se debe a la mayor frecuencia de reloj empleada y a la ausencia de interrupciones provocadas por el chip de video (en el C-64 se producen constantemente para detectar las colisiones de los sprites aunque éstos no se utilicen).

La tabla comparativa de tiempos para este ordenador es:

Elementos	Longitud	Basic	C.M.
10	10	1,1	0,013
100	10	97,3	0,666
1000	40	111,4	0,733

(los tiempos de esta tabla están medidos en segundos).

La prueba con una matriz de 1.000 elementos no la hicimos por falta de memoria, pero quien disponga de una ampliación de RAM de 64 Kbytes como la de Ferre Moret y suficiente paciencia para esperar los resultados de la rutina de ordenación en Basic, podrá realizarla y esperamos que nos diga cuántas horas tarda el Basic frente a los segundos empleados por la rutina de código máquina para ordenar la misma matriz.

Al igual que su gemela para el C-64, esta rutina puede cargarse por cualquiera de los siguientes métodos:

1.—Con la ayuda de un macroensamblador similar al del C-64. Esta opción creo que sólo es posible si empleamos un C-64 para ensamblar, luego pasamos el programa ensamblado a disco y lo conectamos a un Vic-20 para cargar el módulo ejecutable de código máquina. Puede que exista algún ensamblador que permita la utilización de etiquetas y directivas de ensamblado para el Vic, pero yo no lo tengo ni lo conozco.

2.—Con ayuda de un monitor de lenguaje máquina introduciendo en memoria el volcado hexadecimal (listado 2); para ello podemos emplear el VICMON, Tinimon o cualquier otro de los

monitores de lenguaje máquina para el Vic-20 disponibles en el mercado o en los clubs de usuarios.

3.—Tecleando el programa cargador Basic que aparece en el listado 3. En este caso es importante realizar un par de POKes y un NEW antes de teclear el programa Basic. Los Pokes son:

POKE 44,20  
POKE 20\*256,0  
NEW

...ahora ya podemos teclear el listado 3.

Para emplear este programa suponemos que tenemos conectada al menos una ampliación de 8 K RAM, ya que si no la tenemos, los ficheros o programas de gestión de ficheros que realizaremos serán tan pequeños que no merecerá la pena emplear esta rutina de ordenación. De este modo simplificamos bastante la rutina de colocación en memoria, y por otro lado no es lógico gestionar ficheros sin una ampliación de memoria de al menos 8 K.

El programa se coloca automáticamente al final de la memoria disponible, bajando él mismo los punteros de final de memoria y comienzo de cadenas de caracteres. Para ello he colocado una rutina de transferencia de memoria al principio del programa, y



## EXCURSION POR LA MICROPROGRAMACION

para no tener ningún tipo de problema al llamarla, he colocado una línea del tipo 10 SYS 4622 al principio de las rutinas de código máquina que permite llamar la rutina de transferencia con sólo hacer RUN como en cualquier programa Basic.

Al igual que en el C-64, con el programa del listado número 4 podemos comprobar los tiempos de clasificación de la rutina de código máquina para matrices de distintos tamaños, y el listado 5 es un simil Basic del programa de ordenación que sirve para comparar tiempos frente al anterior.

La rutina de clasificación del simil Basic no es la mejor; en realidad es sólo la equivalente al programa de código máquina para poder entender su funcionamiento. Naturalmente el que desee emplear un algoritmo de clasificación más perfeccionado puede hacerlo, y luego deberá pasarlo a lenguaje máquina, pero creo que los tiempos obtenidos con esta sencilla rutina son ya suficientemente buenos para no necesitar otra más rápida.

Naturalmente, las limitaciones que impusimos en la rutina del C-64 se mantienen en ésta y son:

1.—La matriz debe ser de una sola dimensión, por ejemplo: DIM A\$(100)

No podemos emplear esta rutina con matrices multidimensionales como DIM A\$(100,4), ya que el algoritmo de intercambio de elementos sería más complicado, y el empleado sólo intercambia los elementos de matrices de una dimensión.

2.—La matriz debe contener un elemento más de los que necesitamos, y este elemento debe contener un nulo ""; de este modo la rutina comprueba que ha llegado al final de la matriz y no destroza las variables situadas a continuación. Si diseñásemos una rutina que clasificase todos los elementos de la matriz, en el caso de tener una matriz con elementos vacíos, al clasificarla obtendríamos un montón de elementos vacíos en las primeras posiciones, y luego los demás.

3.—Es necesario disponer de alguna ampliación de RAM igual o superior a 8 K (podemos emplear 8, 16, 24, 32 ó 64 sin que ello afecte al funcionamiento del

programa).

Como la rutina se coloca al final de la memoria disponible y ésta no es siempre la misma (depende de la ampliación conectada), no podemos decir un SYS fijo para llamar a la rutina, pero la forma ideal de hacerlo sería:

1) Asignar a unas variables las direcciones de entrada de las rutinas de clasificación y búsqueda haciendo:

AA=256\*PEEK(56)+PEEK(55)

AB=AA+139

2) Para emplear la rutina de clasificación:

SYS AA, A\$(1)

3) Para emplear la rutina de búsqueda:

SYS AB, A\$(1), A\$, A

Voy a explicar el significado de cada una de estas variables, que naturalmente pueden ser cambiadas por otras del mismo tipo.

Al llamar la rutina de ordenación debemos indicar la matriz a ordenar y el número de elemento a partir del cual la deseamos hacer; esto lo hacemos poniendo el nombre de la matriz después de la coma (A\$(x)) en el ejemplo, e indicando el primer elemento a clasificar (el 1 en el caso anterior). Los elementos anteriores al indicado permanecerán intactos en su lugar; esto permite ordenar matrices parcialmente.

Si deseamos ordenar una matriz llamada PR\$(X) a partir de su octavo elemento, la llamada sería:

SYS AA, PR\$(8)

La única condición es que sean matrices de cadenas de caracteres y que el último sea un nulo como dijimos antes.

Para emplear la rutina de búsqueda, debemos especificar la matriz donde buscar, el número de elemento a partir del cual realizar la búsqueda, la cadena de caracteres a buscar y la variable donde deseamos que nos indique la posición del elemento encontrado.

En el ejemplo buscamos la cadena de caracteres contenida en la variable A\$ entre los elementos de la matriz A\$(X) a partir del primero (A\$(1)), y el número de elemento hallado lo guardamos en la variable A.

Un par de detalles importantes: El número devuelto en A es

tres veces el número del elemento encontrado, ya que no hemos querido complicar la rutina de C.M. realizando la división por tres (es mucho más sencillo hacer en Basic A=A/3). Y la segunda es que la rutina se detiene en el elemento siguiente al buscado, es decir, en el primer elemento que sea mayor que el buscado; esto nos obliga a restar uno al resultado y/o mirar el elemento anterior.

Es un requisito indispensable que la matriz en la que vamos a realizar la búsqueda esté previamente ordenada en orden creciente. Esto no será ningún inconveniente, ya que la rutina de clasificación es prácticamente "instantánea".

Todas las limitaciones establecidas para la rutina de ordenación se aplican a esta de búsqueda (que la matriz sea de una única dimensión, etc.); con todo esto simplificamos bastante la rutina.

También debo recalcar que, tanto en la rutina de búsqueda como en la de clasificación, es necesario que los tipos de variables sean los mismos que los empleados en los ejemplos anteriores. No se puede cambiar la variable de coma flotante "A" por otra de tipo entero "A%", pero sí se puede sustituir por otra del mismo tipo como "WE"; esto mismo es aplicable a las demás variables necesarias al llamar las rutinas de código máquina.

En el listado 6 aparece el equivalente Basic a la rutina de búsqueda, si lo estudiáis detenidamente veréis el sencillo algoritmo de búsqueda empleado en la rutina de código máquina.

Espero que estas rutinas reanimen a los usuarios del ordenador Vic-20 de Commodore que lo habían abandonado en el rincón de los trastos cuando se cansaron de no encontrar programas interesantes, o se compraron un modelo superior (C-64 o C-128), ya que viéndolo desde el punto de vista de la velocidad el Vic-20 es superior a todos ellos y en capacidad de memoria no se queda muy atrás si disponemos de una tarjeta de ampliación de 64 K RAM. Visto de este modo es mucho mejor que sus hermanos mayores... ¡Animo! no dejéis

# EXCURSION POR LA MICROPROGRAMACION

arrinconada esa pequeña joya que tiene el mérito de haber sido el primer ordenador personal con teclado serio y preciso asequible que entró en nuestro país, y da mucha pena dejarlo de lado después de tan buenos ratos con él.

He de reconocer que la venilla sentimental por mi Vic-20 la sacó a la luz un radioaficionado (EA4AJJ) de Madrid que un día me dijo por radio que su ordena-

dor se estaba quedando "sólo para recibir RTTY (radio-tele-tipo) y CW (telegrafía)", ya que las rutinas que estaba probando para ordenar un pequeño fichero tardaban horas, y esto le hacía resistir en su intento de aplicarlo a la gestión de ficheros en casette.

En ese mismo instante le dije por la banda de 2 metros que mirase el número 15 de nuestra

revista (de la que es suscriptor), pero me di cuenta de mi error: aquella rutina sólo funcionaba en el C-64. Rápidamente me puse en acción, saqué del armario mi Vic y en diez minutos... rutina de clasificación para el Vic. Desde entonces el Vic ha vuelto a merecer por mi mesa de trabajo junto al C-64 y al IBM-PC/XT; espero que no se retire de ella durante algunos meses...

## Listado 1

2

990: 1201

1040: 1201  
1050: 1201  
1060: 1201  
1070: 1201  
1080: 1201  
1090: 1201  
1100: 1201  
1110: 1201  
1120: 1201  
1130: 1201  
1140: 1201  
1150: 1201  
1160: 1201  
1170: 1201 0C 08 0A  
1180: 1207 34 36 32

1220: 120E C6 38  
1230: 1210 C6 38  
1240: 1212 A9 12  
1250: 1214 A0 38  
1260: 1216 85 FF  
1270: 1218 84 FE  
1280: 121A A0 00  
1290: 121C B1 FE  
1300: 121E 91 37  
1310: 1220 C8  
1320: 1221 D0 F9  
1330: 1223 E6 FF  
1340: 1225 E6 38  
1350: 1227 A5 38  
1360: 1229 C5 34  
1370: 122B D0 EF  
1380: 122D A6 38  
1390: 122F CA  
1400: 1230 CA  
1410: 1231 86 38  
1420: 1233 86 34  
1430: 1235 86 36  
1440: 1237 60

1460: 1238 20 FD CE START  
1470: 123B 20 8B D0  
1480: 123E 85 6A  
1490: 1240 84 6B  
1500: 1242 A0 00  
1510: 1244 B1 6A  
1520: 1246 F0 7A  
1530: 1248 85 6E  
1540: 124A C8  
1550: 124B B1 6A  
1560: 124D 99 6E 00  
1570: 1250 C0 02  
1580: 1252 D0 F6  
1590: 1254 A5 6A

```

.OPT P2
; ESTE PROGRAMA SIRVE PARA ORDENAR
; CADENAS DE CARACTERES DE BASIC
; HACIENDOLO EN CODIGO MAQUINA.
;
FRETOP = $33 ; PUNTEROS INICIO CADENAS
FRESPC = $35 ; PUNTERO TEMPORAL
MEMSIZ = $37 ; FIN DE RAM
ASIGNA = $D000 ; VALOR A UNA VARIABLE
CONVTF = $0391 ; PASA A COMA FLOTANTE
LONGIT = $FF ; LONGITUD MENOR
BUSCOM = $CEFD ; BUSCA COMA
BUSVAR = $D00B ; BUSCA VARIABLE
ELEM1 = $6A ; PRIMER ELEMENTO DE MATRIZ
ELEM2 = $6C ; SEGUNDO ELEMENTO
POINT1 = $6E ; PUNTERO A PRIMER STRING
POINT2 = $FC ; PUNTERO AL SEGUNDO
*= $1201
.BYTE$0C,$08,$0A,$00,$0E,$20
.BYTE$34,$36,$32,$32,$00,$00,$00
; ESTOS DATOS ORIGINAN UNA
; LINEA DE PROGRAMA BASIC
; 10 SYS 4622
DEC MEMSIZ+1 ; BAJA PUNTERO DEL
DEC MEMSIZ+1 ; FIN DE RAM
LDA #>START
LDY #<START ; PONE PUNTEROS PARA
STA $FF ; TRANSFERIR LA RUTINA
STY $FE
LDY #0
LDA ($FE),Y ; COPIA UN BYTE
STA (MEMSIZ),Y
INY ; SIGUIENTE BYTE
BNE TRANSF1
INC $FF ; COMPROBEA SI HA
INC MEMSIZ+1 ; COPIADO DOS PAGINAS
LDA MEMSIZ+1 ; (512 BYTES)
CMP FRETOP+1
BNE TRANSF1
LDX MEMSIZ+1 ; BAJA LOS PUNTEROS
DEX ; DE FIN DE RAM
DEX ; PARA PROTEGER LA
STX MEMSIZ+1 ; RUTINA DE C.M.
STX FRETOP+1
STX FRESPC+1
RTS ; FIN DE RUTINA

;
JSR BUSCOM ; BUSCA LA COMA
JSR BUSVAR ; CALCULA DIRECCION DEL
STA ELEM1 ; PRIMER ELEMENTO DE
STY ELEM1+1 ; LA MATRIZ.
LDY #0 ; ESTO COMPROBEA SI LA
LDA (ELEM1),Y ; LONGITUD DEL PRIMER
BEQ FIN ; STRING ES CERO.
STA POINT1 ; LA GUARDA SI NO ES CERO
INY
LDA (ELEM1),Y
STA POINT1,Y
CPY #2
BNE LI1020
LDA ELEM1 ; AQUI HACE LO MISMO

```



1600  
 1610  
 1620  
 1630  
 1640  
 1650  
 1660  
 1670  
 1680  
 1690  
 1700  
 1710  
 1720  
 1730  
 1740  
 1750  
 1760  
 1770  
 1780  
 1790  
 1800  
 1810  
 1820  
 1830  
 1840  
 1850  
 1860  
 1870  
 1880  
 1890  
 1900  
 1910  
 1920  
 1930  
 1940  
 1950  
 1960  
 1970  
 1980  
 1990  
 2000  
 2010  
 2020  
 2030  
 2040  
 2050  
 2060  
 2070  
 2080  
 2090  
 2100  
 2110  
 2120  
 2130  
 2140  
 2150  
 2160  
 2170  
 2180  
 2190  
 2200  
 2210  
 2220  
 2230  
 2240  
 2250  
 2260  
 2270  
 2280  
 2290  
 2300  
 2310  
 2320  
 2330  
 2340  
 2350  
 2360  
 2370  
 2380  
 2390  
 2400  
 2410  
 2420  
 2430  
 2440  
 2450  
 2460  
 2470  
 2480  
 2490  
 2500  
 2510  
 2520  
 2530  
 2540  
 2550  
 2560  
 2570  
 2580  
 2590  
 2600  
 2610  
 2620  
 2630  
 2640  
 2650  
 2660  
 2670  
 2680  
 2690  
 2700  
 2710  
 2720  
 2730  
 2740  
 2750  
 2760  
 2770  
 2780  
 2790  
 2800  
 2810  
 2820  
 2830  
 2840  
 2850  
 2860  
 2870  
 2880  
 2890  
 2900  
 2910  
 2920  
 2930  
 2940  
 2950  
 2960  
 2970  
 2980  
 2990  
 3000  
 3010  
 3020  
 3030  
 3040  
 3050  
 3060  
 3070  
 3080  
 3090  
 3100  
 3110  
 3120  
 3130  
 3140  
 3150  
 3160  
 3170  
 3180  
 3190  
 3200  
 3210  
 3220  
 3230  
 3240  
 3250  
 3260  
 3270  
 3280  
 3290  
 3300  
 3310  
 3320  
 3330  
 3340  
 3350  
 3360  
 3370  
 3380  
 3390  
 3400  
 3410  
 3420  
 3430  
 3440  
 3450  
 3460  
 3470  
 3480  
 3490  
 3500  
 3510  
 3520  
 3530  
 3540  
 3550  
 3560  
 3570  
 3580  
 3590  
 3600  
 3610  
 3620  
 3630  
 3640  
 3650  
 3660  
 3670  
 3680  
 3690  
 3700  
 3710  
 3720  
 3730  
 3740  
 3750  
 3760  
 3770  
 3780  
 3790  
 3800  
 3810  
 3820  
 3830  
 3840  
 3850  
 3860  
 3870  
 3880  
 3890  
 3900  
 3910  
 3920  
 3930  
 3940  
 3950  
 3960  
 3970  
 3980  
 3990  
 4000  
 4010  
 4020  
 4030  
 4040  
 4050  
 4060  
 4070  
 4080  
 4090  
 4100  
 4110  
 4120  
 4130  
 4140  
 4150  
 4160  
 4170  
 4180  
 4190  
 4200  
 4210  
 4220  
 4230  
 4240  
 4250  
 4260  
 4270  
 4280  
 4290  
 4300  
 4310  
 4320  
 4330  
 4340  
 4350  
 4360  
 4370  
 4380  
 4390  
 4400  
 4410  
 4420  
 4430  
 4440  
 4450  
 4460  
 4470  
 4480  
 4490  
 4500  
 4510  
 4520  
 4530  
 4540  
 4550  
 4560  
 4570  
 4580  
 4590  
 4600  
 4610  
 4620  
 4630  
 4640  
 4650  
 4660  
 4670  
 4680  
 4690  
 4700  
 4710  
 4720  
 4730  
 4740  
 4750  
 4760  
 4770  
 4780  
 4790  
 4800  
 4810  
 4820  
 4830  
 4840  
 4850  
 4860  
 4870  
 4880  
 4890  
 4900  
 4910  
 4920  
 4930  
 4940  
 4950  
 4960  
 4970  
 4980  
 4990  
 5000  
 5010  
 5020  
 5030  
 5040  
 5050  
 5060  
 5070  
 5080  
 5090  
 5100  
 5110  
 5120  
 5130  
 5140  
 5150  
 5160  
 5170  
 5180  
 5190  
 5200  
 5210  
 5220  
 5230  
 5240  
 5250  
 5260  
 5270  
 5280  
 5290  
 5300  
 5310  
 5320  
 5330  
 5340  
 5350  
 5360  
 5370  
 5380  
 5390  
 5400  
 5410  
 5420  
 5430  
 5440  
 5450  
 5460  
 5470  
 5480  
 5490  
 5500  
 5510  
 5520  
 5530  
 5540  
 5550  
 5560  
 5570  
 5580  
 5590  
 5600  
 5610  
 5620  
 5630  
 5640  
 5650  
 5660  
 5670  
 5680  
 5690  
 5700  
 5710  
 5720  
 5730  
 5740  
 5750  
 5760  
 5770  
 5780  
 5790  
 5800  
 5810  
 5820  
 5830  
 5840  
 5850  
 5860  
 5870  
 5880  
 5890  
 5900  
 5910  
 5920  
 5930  
 5940  
 5950  
 5960  
 5970  
 5980  
 5990  
 6000  
 6010  
 6020  
 6030  
 6040  
 6050  
 6060  
 6070  
 6080  
 6090  
 6100  
 6110  
 6120  
 6130  
 6140

2210:	1203	20	FD	CE
2220:	1206	20	8B	00
2230:	1209	85	6A	
2240:	120B	84	6B	
2250:	120D	8D	3C	03
2260:	1200	8C	30	03
2270:	1203	20	FD	CE
2280:	1206	20	8B	00
2290:	1209	85	6C	
2300:	120B	84	6D	
2310:	120D	A0	00	
2320:	120F	B1	6A	
2330:	12E1	F0	47	
2340:	12E3	85	6E	
2350:	12E5	C8		

LAZ081

# EXCURSION POR LA MICROPROGRAMACION

```

2360: 12E6 B1 6A
2370: 12E8 99 6E 00
2380: 12E8 C0 02
2390: 12ED 00 F6
2400: 12EF A0 00
2410: 12F1 B1 6C
2420: 12F3 F0 35
2430: 12F5 85 FC
2440: 12F7 C8
2450: 12F8 B1 6C
2460: 12FA 99 FC 00
2470: 12FD C0 02
2480: 12FF 00 F6
2490: 1301 A5 FC
2500: 1303 C5 6E
2510: 1305 90 02
2520: 1307 A5 6E
2530: 1309 85 FF
2540: 130B A0 00
2550: 130D B1 6F
2560: 130F 01 FD
2570: 1311 F0 04
2580: 1313 90 07
2590: 1315 B0 13
2600: 1317 C8
2610: 1318 C4 FF
2620: 131A 00 F1
2630: 131C 18
2640: 131D A5 6A
2650: 131F 69 03
2660: 1321 85 6A
2670: 1323 90 02
2680: 1325 E6 68
2690: 1327 18
2700: 1328 90 B3
2710: 132A 38
2720: 132B A5 6A
2730: 132D ED 3C 03
2740: 1330 85 6A
2750: 1332 A5 68
2760: 1334 ED 3D 03
2770: 1337 85 68
2780: 1339 20 FD CE
2790: 133C 20 88 00
2800: 133F 85 49
2810: 1341 84 4A
2820: 1343 A5 68
2830: 1345 A4 6A
2840: 1347 20 91 03
2850: 134A 20 00 0B
2860: 134D 68

LAZOB3
LDY #0 ; COPIA LOS PUNTEROS
LDA (ELEM2),Y ; DEL STRING BUSCADO
BEQ ENCONT ; SI LEN(A$)=0-->FIN
STA POINT2

LAZOB4
INY
LDA (ELEM2),Y
STA POINT2,Y
CPY #2
BNE LAZOB4
LDA POINT2 ; LONGITUD DEL BUSCADO
CMP POINT1 ; LONGITUD DEL OTRO
BCC MAYORL
LDA POINT1
STA LONGIT ; LONGITUD A COMPARAR
LDY #0
LDA (POINT1+1),Y ; COMPARA UNA
CMP (POINT2+1),Y ; LETRA
BEQ SIGBUS ; SIGUIENTE LETRA
BCC SIGELE ; SIGUIENTE ELEMENTO
BCS ENCONT ; SI ENCONTRADO

MAYORL
LDY #0
LDA (POINT1+1),Y ; COMPARA UNA
CMP (POINT2+1),Y ; LETRA
BEQ SIGBUS ; SIGUIENTE LETRA
BCC SIGELE ; SIGUIENTE ELEMENTO
BCS ENCONT ; SI ENCONTRADO

SIGBUS
INY
CPY LONGIT
BNE LAZOB5 ; SIGUIENTE ELEMENTO
LDA ELEM1 ; DE LA MATRIZ
ADC #3
STA ELEM1
BCC NOARRA
INC ELEM1+1

SIGELE
CLC
LDA ELEM1 ; SIGUIENTE ELEMENTO
ADC #3
STA ELEM1
BCC NOARRA
INC ELEM1+1

NOARRA
CLC
BCC LAZOB1
SEC
LDA ELEM1 ; CALCULA EL NUMERO
SBC #033C ; DEL ELEMENTO ENCONTRADO
STA ELEM1 ; O EL ULTIMO SI LA
LDA ELEM1+1 ; BUSQUEDA HA FALLADO
SBC #033D
STA ELEM1+1
JSR BUSCOM
JSR BUSVAR
STA $49
STY $4A
LDA ELEM1+1
LDY ELEM1
JSR CONVTF ; PASA A COMA FLOTANTE
JSR ASIGNA ; LO ASIGNA A LA VARIABLE
RTS

```

## Listado 2

```

.:1200 00 0C 08 0A 00 9E 20 34
.:1208 36 32 32 00 00 00 C6 38
.:1210 C6 38 A9 12 A0 38 85 FF
.:1218 84 FE A0 00 B1 FE 91 37
.:1220 C8 00 F9 E6 FF E6 38 A5
.:1228 38 C5 34 00 EF A6 38 CA
.:1230 CA 86 38 86 34 86 36 60
.:1238 20 FD CE 20 88 00 85 6A
.:1240 84 68 A0 00 B1 6A F0 7A
.:1248 85 6E C8 B1 6A 99 6E 00
.:1250 C0 02 00 F6 A5 6A 85 6C
.:1258 A5 68 85 6D 18 A5 6C 69
.:1260 03 85 6C 90 02 E6 6D A0
.:1268 00 B1 6C F0 47 85 FC C5
.:1270 6E 90 02 A5 6E 85 FF C8
.:1278 B1 6C 99 FC 00 C0 02 00
.:1280 F6 A0 00 B1 6F 01 FD F0
.:1288 04 90 01 B0 09 C8 C4 FF
.:1290 00 F1 C4 6E 00 C6 A0 00
.:1298 B1 6A AA B1 6C 91 6A 99
.:12A0 6E 00 8A 91 6C C8 C0 03
.:12A8 00 EE A0 00 B1 6A 85 FF
.:12B0 A9 00 F0 A8 18 A5 6A 69
.:12B8 03 85 6A 90 02 E6 68 18
.:12C0 90 80 60 20 FD CE 20 88
.:12C8 00 85 6A 84 68 80 3C 03
.:12D0 8C 3D 03 20 FD CE 20 88
.:12D8 00 85 6C 84 6D A0 00 B1
.:12E0 6A F0 47 85 6E C8 B1 6A
.:12E8 99 6E 00 C0 02 00 F6 A0
.:12F0 00 B1 6C F0 35 85 FC C8
.:12F8 B1 6C 99 FC 00 C0 02 00
.:1300 F6 A5 FC C5 6E 90 02 A5
.:1308 6E 85 FF A0 00 B1 6F 01
.:1310 FD F0 04 90 07 80 13 C8
.:1318 C4 FF 00 F1 18 A5 6A 69
.:1320 03 85 6A 90 02 E6 68 18
.:1328 90 83 38 A5 6A ED 3C 03
.:1330 85 6A A5 68 ED 3D 03 85
.:1338 68 20 FD CE 20 88 00 85
.:1340 49 84 4A A5 68 A4 6A 20
.:1348 91 03 20 00 0B 60 41 20

```



# EXCURSION POR LA MICROPROGRAMACION

## Listado 3

```

1 GOTO 5
2 : CARGADOR BASIC DEL PROGRAMA
3 : CLASIFICACION Y BUSQUEDA VIC-20
4 <C>1985 DIEGO ROMERO/COMMODORE WORLD
5 LI=100:LF=335:RO=4609:GOSUB 15
6 POKE 44,18:POKE 46,20
7 PRINT"HACER SAVE PARA GUARDARLO"
8 END
9 :
10 :
11 :
12 :
13 :
14 :
15 FOR LN=LI TO LF STEP 5
16 : SU=0:PRINT"LINEA";LN
17 : FOR Z=1 TO 7
18 :   READ DA:SU=SU+DA
19 :   POKE AD,DA:DA=DA+1
20 :   NEXT:READ CS:IF CS<>SU THEN 24
21 NEXT LN
22 RETURN
23 :
24 PRINT"ERROR EN LOS DATOS"
25 PRINT"DE LA LINEA";LN
26 END
100 DATA 12,8,10,0,158,32,52,272
105 DATA 54,50,50,0,0,198,352
110 DATA 56,198,56,169,18,160,56,713
115 DATA 133,255,132,254,160,0,177,1111
120 DATA 254,145,55,200,208,249,230,1341
125 DATA 255,230,56,165,56,197,52,1011
130 DATA 208,239,166,56,202,202,134,1207
135 DATA 56,134,52,134,54,96,32,558
140 DATA 253,206,32,139,208,133,106,1077
145 DATA 132,107,160,0,177,106,240,922
150 DATA 122,133,110,200,177,106,153,1001
155 DATA 110,0,192,2,208,246,165,923
160 DATA 106,133,108,165,107,133,109,861
165 DATA 24,165,108,105,3,133,108,646
170 DATA 144,2,230,109,160,0,177,822
175 DATA 108,240,71,133,252,197,110,1111
180 DATA 144,2,165,110,133,255,200,1009
185 DATA 177,108,153,252,0,192,2,884
190 DATA 208,246,160,0,177,111,209,1111
195 DATA 253,240,4,144,209,176,9,1035
200 DATA 200,196,255,208,241,196,110,1406
205 DATA 176,198,160,0,177,106,170,987
210 DATA 177,108,145,106,153,110,0,799
215 DATA 138,145,108,200,192,3,208,994
220 DATA 238,160,0,177,106,133,255,1069
225 DATA 169,0,240,168,24,165,106,872
230 DATA 105,3,133,106,144,2,230,723
235 DATA 107,24,144,128,96,32,253,784
240 DATA 206,32,139,208,133,106,132,956
245 DATA 107,141,60,3,140,61,3,515
250 DATA 32,253,206,32,139,208,133,1003
255 DATA 108,132,109,160,0,177,106,792
260 DATA 240,71,133,110,200,177,106,1037
265 DATA 153,110,0,192,2,208,246,911
270 DATA 160,0,177,108,240,53,133,871
275 DATA 252,200,177,108,153,252,0,1142
280 DATA 192,2,208,246,165,252,197,1262
285 DATA 110,144,2,165,110,133,255,919
290 DATA 160,0,177,111,209,253,240,1150
295 DATA 4,144,7,176,19,200,196,746
300 DATA 255,208,241,24,165,106,105,1104
305 DATA 3,133,106,144,2,230,107,725
310 DATA 24,144,179,56,165,106,237,911
315 DATA 60,3,133,106,165,107,237,811
320 DATA 61,3,133,107,32,253,206,795
325 DATA 32,139,208,133,73,132,74,791
330 DATA 165,107,164,106,32,145,211,930

```

## Listado 4

```

100 OPEN4,4:AA=256*PEEK(56)+PEEK(55)
105 AB=AA+139
110 INPUT"NUMERO, LONGITUD";N,L
120 DIM A$(N):N=N-1
130 FOR I=0 TO N
140 FOR J=1 TO RND(0)*L
150 A$(I)=A$(I)+CHR$(RND(0)*59+32)
160 NEXT:NEXT
170 FOR I=0 TO N:PRINTA$(I):NEXT
180 PRINT#4:CLOSE4
190 TI$="000000":SYS AA,A$(0):T$=TI$
200 OPEN4,4
210 PRINT#4,:PRINT#4,"TIEMPO ORDENACION"
= "TI$HMMSS":PRINT#4,:PRINT#4,
220 FORI=0TON:PRINTA$(I):NEXT
230 PRINT#4:CLOSE4
240 REM DEMOSTRACION DE BUSQUEDA
250 INPUT "STRING A BUSCAR";A$
260 SYS AB,A$(0),A$,A
270 A=A/3
280 PRINT"NUMERO DE ELEMENTO=";A
290 PRINT"ELEMENTO A$(");A;"=";A$(A)
300 IF A>0 THEN PRINT"ELEMENTO A$(");A-1;
="";A$(A-1)
310 GOTO240

```

## Listado 5

```

100 OPEN4,4
110 INPUT "NUMERO, LONGITUD";N,L
120 DIM A$(N):N=N-1
130 FOR I=0 TO N
140 FOR J=1 TO RND(0)*L
150 A$(I)=A$(I)+CHR$(RND(0)*59+32)
160 NEXT:NEXT
170 FOR I=0 TO N:PRINT#4,A$(I):NEXT
180 PRINT#4:CLOSE4
190 TI$="000000":GOSUB1000:T$=TI$
200 OPEN4,4
210 PRINT#4,:PRINT#4,"TIEMPO ORDENACION"
= "TI$HH.MM.SS.":PRINT#4,:PRINT#4,
220 FORI=0TON:PRINT#4,A$(I):NEXT
230 PRINT#4:CLOSE4
240 END
1000 I=0
1010 IFA$(I)=""THENRETURN
1020 J=I
1030 J=J+1
1040 IFA$(J)=""THEN1070
1050 IFA$(I)>A$(J)THENB$=A$(I):A$(I)=A$(J):A$(J)=B$
1060 GOTO1030
1070 I=I+1:GOTO1010

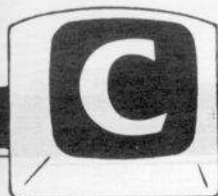
```

## Listado 6

```

10 REM EQUIVALENTE PARA A$(N),A$,A
20 N=0
30 IFA$(N)=""THEN 50
40 IFA$(N)<A$ THEN N=N+1:GOTO30
50 A=N
60 END

```



## Platine 64

Ordenador Commodore 64  
o C-128 en modo 64.  
Con unidad de disco 1541  
e impresora o plotter  
Precio: 30.000 ptas.  
Fabricante: Data Becker  
Distribuidor: Ferré-Moret S.A.  
Tusset, 8, Entresuelo, 2  
08006 Barcelona  
Teléfono: 93-218 02 93

Este programa da al Commodore 64 la posibilidad de ser empleado para el diseño de placas de circuito impreso a doble cara.

Hasta ahora sólo se podían emplear ordenadores profesionales de alto precio, con la ayuda de un programa de mayor coste que el propio ordenador (el programa para el IBM-PC/XT cuesta de 1 a 4 millones de pesetas) para diseñar este tipo de circuitos impresos. Naturalmente esas cantidades son totalmente prohibitivas para el aficionado medio.

Hace unos meses la casa alemana Data Becker lanzó al mercado este programa para el Commodore 64, dándole la potencia de ordenadores profesionales a nuestro ordenador personal, y poco después la versión en español del programa estaba disponible en su distribuidor para España que es Ferré-Moret de Barcelona.

Con este programa es posible emplear tanto la impresora Commodore MPS-801 como plotters (trazadores gráficos) de distintos tipos, pudiendo ser estos últimos autoconstruidos ya que los comandos o secuencias de control del plotter pueden ser alterados en el programa según las necesidades del usuario.

Aquellos que han intentado alguna vez diseñar un circuito impreso sabrán que no es tarea sencilla ni rápida; por el contrario exige mucha paciencia y seguir un método de trabajo ordenado, ya que de lo contrario muchas veces se nos olvidarán las conexiones de algunos componentes del circuito; además cuando dos pistas se cruzan se suele dar un rodeo con una de ellas, pero cuando son más la cosa se complica hasta el punto de tener que redistribuir todos los componentes y volver a comenzar el diseño del trazado de las pistas. Esta complicación crece si se trata de diseñar placas que contienen líneas de bus de datos y/o direcciones entre los distintos chips (en el caso del hard de ordenadores).

Las fases más pesadas del diseño son dos: Poner las pistas de la manera más ventajosa posible y realizar el dibujo de la manera más precisa que podamos. En estos dos pasos es donde nos ayuda el programa Platine 64.

El programa diseña automáticamente el trazado de las pistas del circuito gracias a la función "autorouter", y en caso necesario podemos rectificar o el diseño manualmente por medio del "manualrouter".

Las fases del diseño con el programa son: 1.—Decirle al programa qué componentes vamos a emplear; para ello contestamos a una serie de preguntas que nos hace el programa.

Cuando los componentes tienen el patillaje igual no es necesario repetirlo, simplemente



se lo indicamos al programa. Además podemos guardar el trabajo realizado en el disco para formar una biblioteca de componentes para futuros circuitos.

2.—Indicar la colocación de componentes que deseamos.

3.—Realizar una tabla de conexiones siguiendo el esquema eléctrico, por ejemplo: patilla 2 de circuito integrado número 1 unido al colector del transistor TR3.

En este paso se simplifica enormemente el proceso cuando se emplean circuitos integrados del tipo empleado en ordenadores, ya que las uniones de todas las patillas del bus se realizan en conjunto, y el programa se encarga de trazar todas las líneas de datos o direcciones correspondientes.

Las conexiones realizadas pueden ser grabadas también en el disco para su posterior utilización o rectificación.

4.—En este momento llamamos a la función "Autorouter" que es la encargada de intentar realizar todas las conexiones entre todos los componentes del circuito.

Decimos intentar realizar porque no siempre es posible realizarlas con una determinada distribución de componentes. Algunas veces el programa se detiene cuando no es

capaz de trazar las pistas del circuito al cruzarse éstas; en este caso la solución es cambiar la posición de algún componente conflictivo y volver a intentarlo.

En un 90% de los casos el éxito es total a la primera, es decir, se consigue el diseño completo del circuito sin necesidad de cambiar la distribución de los componentes en la placa. En los restantes casos puede ser necesario cambiar una pista de posición con el "manualrouter" para facilitar la continuación del proceso al programa.

5.—El último paso es imprimir el circuito impreso diseñado con ayuda de la impresora Commodore MPS 801.

Quien disponga de un plotter podrá obtener directamente el dibujo del circuito impreso en papel vegetal.

Un detalle que antes hemos mencionado y que interviene en este paso es que el plotter puede ser autoconstruido, ya que los comandos o secuencias de caracteres de control para moverlo se pueden definir en una de las opciones del programa.

Finalmente, este programa tiene más de cien funciones diferentes, todas ellas están controladas por menú, no siendo necesario memorizar el modo de empleo. Todo el programa y el manual ha sido traducido al castellano, por lo que tampoco es necesario saber alemán o inglés para manejarlo.

La limitación de memoria del C-64 es la que impide el éxito total en el 100% de los casos de diseño de circuitos, imponiendo al mismo tiempo la limitación del tamaño de las placas de circuito impreso al formato "Eurocard" de doble cara (110x160 mm.).

El programa permite emplear tanto la conexión original del bus serie IEEE de Commodore como las conexiones estándar RS-232 o Centronics para controlar el plotter.

El programa será de gran utilidad a aquellos aficionados a la electrónica que deseen emplear su ordenador para diseñar sus circuitos sin tener que invertir una gran cantidad de dinero en un ordenador más profesional ni un programa caro.

## Hardball

Fabricante: Accolade  
Ordenador: C-64 o C-128 (modo 64)  
Importador:  
Compulogical  
Sta. Cruz de Marcenado, 31  
28015 Madrid  
Teléf.: (91) 241 10 63  
Precio: 2.300 ptas.

Vaya, vaya! Esto es como el anuncio de colonia: "Hay otros juegos de béisbol, pero están en ti". Concretamente hemos tenido noticias desde los EE.UU. de otros juegos de béisbol, de Gamestar, de Epyx, y el World Series Baseball. No hablaré de ellos aquí más que para señalar qué novedades aporta este Hardball.

Aunque sea difícil, intentaré explicar este programa, metiéndome lo menos posible en







detalles del deporte del béisbol. Quizás aquí esté el problema: el béisbol es un deporte muy poco conocido en este país, pero eso no impide que en su país de origen sea un verdadero fenómeno social, tanto o más importante como podrían ser los toros y el fútbol aquí. Aunque últimamente está decayendo y cediendo su sitio ante otros deportes de acción más trepidante y continua como el baloncesto o el fútbol americano. Pero el béisbol sigue teniendo "algo" casi mágico que le mantiene allí. Como deporte, sin duda por su aspecto técnico es bastante singular; los entrenadores de los equipos de béisbol son auténticos "catedráticos" en su materia. El béisbol es de los juegos más complicados de jugar que conozco (esto no significa que sea excepcionalmente difícil jugar al Hardball), ya que sólo los jugadores de las ligas mayores (algo así como la primera división, pero en béisbol) son capaces de hacer cosas como lanzar bolas curvas a la velocidad que las lanzan, coger pelotas "in extremis" tirándose al suelo, etc., y que se pueden realizar en este programa con tan sólo apretar el botón del joystick. Además, sus especiales características como puede ser la especialización de sus jugadores, la emoción controlada que se concentra en unos cuantos momentos selectos, después de una elaboración lenta y metódica (eso es, como en una película de suspense) dan más interés al juego.

Es esta última parte, no tanto perteneciente al juego en sí, en la que los diseñadores nos han querido ofrecer el poder actuar como entrenador de la plantilla completa, con un banquillo surtido, como en los equipos de verdad. Otra cosa que el programa también reproduce es el sistema de medias personales de cada jugador (otro invento yanqui, cómo no) que nos sirve para darnos una idea de qué nivel, qué tipo de bateo, qué características de juego tiene cada jugador. Como en las instrucciones del juego en español no viene nada de eso y dicho sea de paso son una grandísima chapuza (¡estoy indignado, esto es el colmo!), les echaremos una mano a estos muchachos de Compulogical con este pequeño glosario de términos y siglas que aparecen en el juego (si a alguien le sirve de algo).

**Tipo de bateo:** L(left), R(right) o S(lider). Sin comentarios.

**Posición:** 1B, 2B, 3B, SS (short-stop), LF, CF, RF (Left, Right y Center Fielder; "fielder" = jardinero), C(Catcher=recogedor), P(pitcher=lanzador), OF(bateador suplente).

**Promedio de bateo o "Average":** Un número del .000 al .999. Cuanto más alto mejor es el jugador bateando. Valores normales son: menos de .200=malo; entre .200 y .250=malillo; entre .250 y 300=bueno y AB: Siglas de "At Bat". Número de veces que ha bateado en ligas mayores en esta temporada.

**RBI:** Siglas de "Runs Batted In". Número de carreras que se han producido a consecuencia de un hit del bateador.

**HR:** Home Run. Pelotazo fuera de los límites del campo. Número de ellos que ha bateado en lo que lleva de temporada.

**SB:** Siglas de "Stolen base". Número de bases robadas por este bateador en esta temporada. Indicador de si el jugador corre o no.

**R:** "Runs". Número de carreras del partido.

**H: Hits.** Pelotazos "seguros" del bateador, que le permite a él llegar por lo menos a la primera base y a todos los compañeros suyos que ya estaban en base avanzar por lo menos una base.

**E:** Errores cometidos por el equipo y que han permitido a un bateador llegar a base en un caso en que normalmente no llegaría.

**E.R.A.:** Sistema de promedio de los lanzadores. Cuanto más alto mejor.

La publicidad de este programa alardea de "... el juego de béisbol más real de todos los tiempos...", o algo así. En efecto, es tan real, tan real que lo estropea un poco. Primero habría que decir que Hardball pretende simular el béisbol que se juega en las grandes ligas. Pero el nivel que se ha alcanzado en estos últimos veinte años ha llevado a un tipo de juego muy defensivo (ha pasado como en

el fútbol aquí, donde raras veces se ven aquellos "Bazán, 9 — Hércules, 0" de antaño). Así en las ligas mayores, y especialmente en la N.L., los lanzadores (pitchers) son tan buenos que se consigue batear la pelota muy pocas veces. Además, cuando se le da, los jugadores del rombo (infield) que cogen las pelotas bateadas están tan entrenadísimos que casi nunca cometen errores al cogerla o lanzarla a las bases. Esto le da al béisbol ese carácter monótono que en gran parte le ha hecho decaer en los "ranking" de asistencia de público. En Hardball pasa igual. Es muy difícil golpear la pelota. Lo digo por experiencia y no es porque sea un mal jugador de Hardball, ya que he conseguido superar (o casi) el nivel del ordenador, y aún así, el tanteo final nunca pasa de 3-2, 3-1 ó 2-0, con nunca más de ocho o nueve hits por equipo.

## Prontuario del C-64

Ordenador: C-64

Edita:

Editorial NORAY, S. A.

San Gervasio de Cassolas, 79

08022 Barcelona

Teléf.: (93) 211 11 46

Precio: 375 ptas.

Estas "páginas" son, como su propio nombre indica, una guía rápida de todo lo que un programador puede necesitar cuando está trabajando con el ordenador. Todos los "pokes", comandos, tablas ASCII, etc., están incluidos aquí. Es similar al libro **Claves para el C-64** cuya reseña apareció el mes pasado, pero a un nivel más para principiantes.

Conviene destacar la presentación. En formato acordeón, se despliega con facilidad y consta de seis "hojas" escritas por las dos caras con un tipo de letra pequeña pero perfectamente legible. Además, está plastificado y resulta difícil que se rompa, a la vez que gana bastante en presentación.

Al principio vienen una serie de convenciones que se utilizan a lo largo del folleto. Después hay una explicación de cada uno de los comandos y funciones Basic y cómo se utilizan. También habla un poco sobre las variables y los operadores Basic.

Los operadores lógicos (AND, OR, NOT) y las "tablas de la verdad" vienen a continuación. También hay otra tabla con las equivalencias entre los códigos de la pantalla y los códigos ASCII, algo que siempre desconcierta a todo el mundo.

En la parte de atrás del folleto comienza la programación avanzada. Hay un mapa de memoria por bloques (el espacio no da para más) y algo que no indican la mayoría de los manuales: los cambios en los bancos de memoria que puede utilizar el Commodore 64. Después de los códigos de control (los colores, movimientos del cursor, etc.) y las equivalencias —parecido a lo que hacemos

EDITORIAL NORAY, S.A.

## PRONTUARIO DEL

## COMMODORE 64

PALABRAS CLAVE

COMANDOS OPERATIVOS

COMANDOS GRAFICOS Y DE SONIDO

COMANDOS DE COLOR

COMANDOS DE ENTRADA/SALIDA

INSTRUCCIONES BASIC

FUNCIONES BASIC

OPERADORES LOGICOS

MENSAJES DE ERROR

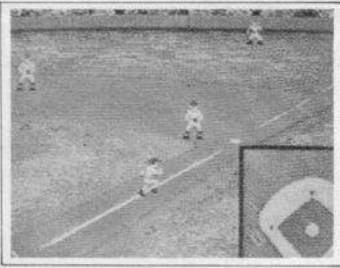
DATOS DE INTERES

**TODO LO QUE NECESITA AL ALCANCE DE LA MANO**

nosotros con la "clave para interpretar los listados"— hay un apartado sobre el modo bit-map y una descripción de los registros del chip Vic. También hay una rápida guía sobre cómo se utilizan los sprites, ya sean normales o en modo multicolor.

Lo que se describe con más profundidad es la utilización de los comandos de sonido, incluyendo todos los pokes que hay que hacer para conseguir sacar algo de sonido del C-64. También describe todas las virguerías de que dispone el chip SID que controla el sonido del ordenador. Lo último de todo el folleto es una descripción de los mensajes de error.

Se echa en falta (aunque el espacio no da para más) algo sobre código máquina e información sobre los periféricos. Sabemos que existen guías similares para otros ordenadores. En definitiva: una buena guía a un buen precio.



De todas maneras esto es un inconveniente menor. Desde luego el realismo del juego se extiende más allá. Se reproducen aquí las características más importantes del juego en las ligas mayores, como el cansancio del lanzador, los robos de base exactamente como son, jugadas de sacrificio aéreo (Sac) o toques de bola (Bunt), doble play, etc. Una cosa que aquí sí se echa en falta son los tiros del lanzador a bases para evitar el robo de bases, así como la posibilidad de hacer retroceder un corredor no forzado a correr (es decir cuando no hay ningún otro corredor detrás).

Vayamos ya directamente a lo que es el programa en sí.

El programa consta de cuatro pantallas y una pequeña "ventana" auxiliar que aparece en dos de estas pantallas. Al cargar el programa aparece también un gráfico muy bonito, cómo no. No cuento aquí el menú inicial en que se elige si se juega contra el ordenador, en casa o fuera, etc.

La primera pantalla es la del entrenador, en la que podemos tomar "esa" decisión estratégica que puede cambiar la historia... del partido. No es ninguna maravilla artística, no tenía que serlo, pero es muy eficaz y completa; un truco: si quieres examinar el banquillo y/o el equipo titular sin tener que hacer un irremediable cambio (si haces un cambio, no podrás recuperar al jugador que has cambiado), escoge la opción de correspondiente y haz como si fueras a cambiar un jugador, pero sin apretar el botón del joystick, y entonces vuelve a la pantalla de la batería para evitar el posible cambio.

La siguiente pantalla es, sin duda, lo más espectacular a todos, es la pantalla de la batería: en el argot, se llama "batería" al conjunto formado por el lanzador y el recogedor (pitcher y catcher) del mismo equipo, y el bateador del equipo contrario. La variedad de lanzamientos es bastante completa: un lanzador concreto posee un "repertorio" limitado de lanzamientos. Un lanzador siempre tiene ventaja sobre un bateador del lado contrario. El recogedor dirige el lanzamiento del lanzador con la posición de su mascota (guante), etc. Todo esto queda reflejado en esta pantalla, hasta el "grito" cadenciado del árbitro (umpire, con U). Esta es la pantalla que más convence de que este programa no se limita a simular los jugadores con monigotes baratos de dos caracteres redefinidos y la pelota con un gráfico de 4x4 pixels. Hay que fijarse también en la indumentaria de los jugadores, el movimiento de bateo, el de lanzamiento, etc...; aquí también "chapeau".

Las dos pantallas restantes no son tan



espectaculares por una simple razón, el terreno que tienen que abarcar. Esto se refleja, sobre todo, en los jugadores de campo, los del fondo, en que se nota que algo no cuadra en la perspectiva. Además cuando se tira una pelota desde estos jugadores a las bases, la pelota sale muy lenta; esto no es un fallo del programa en sí; si se querían obtener gráficos

grandes y bonitos, había que sacrificar algo de exactitud en la relación de tamaños, el "extraño" que hace la pelota es para reflejar la distancia real que existe en el campo, además de que cuanto más grande es la distancia de lanzamiento más lenta y más bombeada tiene que ir la pelota. El movimiento de los corredores, los "plongeon" de los jugadores de campo a por la pelota, los "barridos" (tirarse al suelo) de los corredores al llegar a base, todo eso está muy bien.

Muy buenas horas de diversión me esperan con los campeonatos de Hardball que pienso organizar con los compañeros del equipo (dicho sea de paso, el entrenador, un señor de cincuenta años es el más jugón, para que luego digan). Este programa no puede faltar en la programoteca de los aficionados a los juegos deportivos.

Firmado: un fanático del béisbol.

## Revs, tan real como la vida misma

Ordenador: C-64, C-128

Autor: Geoffrey J. Crammond

Fabricante: Firebird

Distribuye: Serma

Bravo Murillo, 377, 3.ª A

28020 Madrid

Precio: 5.000 ptas.

Tienes enfrente de ti a Johnny Turbo, y tan sólo queda una vuelta para el final de la carrera. Agarras el volante con firmeza y te preparas para adelantarte en la próxima curva. ¡Imposible! Ha hecho un quiebro y casi te saca de la pista. Ahora es tu oportunidad: la curva Woodcote antes de la línea de meta. Pisa el acelerador, cambia de marcha y prepárate para el adelantamiento. ¡Demasiado tarde! Has tomado la curva demasiado abierta y te ha derrapado el coche... Ves cómo todo da vueltas, te elevas al chocar contra el bordillo

y te estrellas contra las vallas de seguridad. Aquí acaba la carrera para ti. O no. Puedes comenzar a correr con sólo pulsar la tecla F1.

Esto es REV's, el simulador de Fórmula-3 más real que he visto. Su nombre viene (creo) de la operación de arranque del coche: aumentar las revoluciones del motor y meter la primera marcha. Básicamente es como los demás juegos de conducción. Se parece mucho a **Pole Position** (al videojuego original, no a la versión para Commodore, que es un poco malilla).

La principal característica de este juego es el parecido con la realidad. Yo nunca he conducido un Fórmula-3, pero comparando este juego con otros existentes en el mercado, la diferencia es abismal. Se encuentra al nivel de **Flight Simulator II** y **Jet** (SubLogic), pero en el plano automovilístico.

El programa, en cinta, viene acompañado de un curioso manual. Además de las instrucciones de rigor hay explicaciones sobre sistemas de conducción, aerodinamismo, cómo tomar las curvas en plan "profesional", etc. Todo ello está acompañado por fotos y dibujos; realizado a conciencia (no se puede decir lo mismo de las traducciones o diseños de otros manuales) porque el programador del juego se documentó ayudándose de David Hunt, un piloto auténtico de Fórmula 3. Además de esto hay un póster con el plano del circuito de Silverstone y de Brands Hatch.

Al comenzar a jugar se puede elegir entre practicar, competir, dar vueltas de clasificación, etc. También hay que señalar el tipo de control que se va a usar. Hay varios: teclado, paddle o joystick analógico. En principio lo más recomendable es utilizar el teclado, porque el utilizar un paddle y un joystick a la vez resulta un poco complicado. Además, los joysticks analógicos (tipo Apple) no los tiene casi nadie.

Desde el teclado puede controlarse el volante, el estérter para arrancar el motor, la palanca de cambios, los frenos, el acelerador y algunos otros comandos para controlar el volumen, volver a comenzar la carrera, etcétera.

Cuando se está en la pista se observan





varias cosas interesantes: el salpicadero del automóvil no es tan espectacular como el de otros juegos (*Skyfox* o *Flight Simulator II*, por citar algunos) sino que contiene solamente lo necesario. No hay contador de velocidad, y en su lugar —el centro de la pantalla— está el cuentarrevoluciones, un instrumento que hay que controlar muy bien. El volante tiene una señal vertical que indica hacia que lado está girando, ya que no basta con moverlo un poquito y que vuelva a su sitio a continuación, sino que hay que mantenerlo equilibrado en todo momento.

Como curiosidades están las ruedas delanteras, que se mueven hacia los lados cuando el coche está girando y los espejos retrovisores que muestran (en pequeñito) lo que hay tras el bólido. Hasta aquí lo que se ve. Hacer que funcione es algo muy distinto.

En primer lugar, hay que colocar la palanca de cambios en punto muerto. A continuación comenzar a acelerar. El ruido de fondo va aumentando y se nota cómo el cuentarrevoluciones también. En el momento oportuno hay que meter primera y el coche comenzará a moverse. Con cada cambio de marcha, la aguja del cuentarrevoluciones pegará unos saltos bruscos hasta estabilizarse. Es importante estudiar este aspecto del coche para comprender mejor su funcionamiento.

Cuando el coche está en movimiento, comienza la "aventura" de tomar las curvas y evitar a los demás coches. Es difícil salir vivo cuando se conduce en tercera o cuarta porque el coche derrapa con mucha facilidad. Esto es un factor regulable al principio de la partida, en la que se pide el ángulo de inclinación del alerón trasero. También es variable el viento lateral que recibe el coche y que le hace salirse de la pista, una pista que está bordeada por unas barreras de seguridad contra las que se estrella el coche a poco que se salga.

Este coche puede salirse "de verdad" de la pista, no como en *Pole Position*, *Pitstop II* —que supera en "emoción" a *Revs*— o de una manera más descarada en *Racing Construction Set*, donde los coches toman las curvas casi solos. Esto es debido a la aparente "dificultad" (en la programación) de que el coche esté rondando por la hierba sin entrar en la pista. En *Revs* no, aquí el coche puede ir por donde quiera y en el sentido que quiera —"contra corriente" incluso— mientras no se choque contra las barreras de protección. También puede ir marcha atrás, lo cual no se ha visto en ningún otro programa todavía.

Como programa es mucho mejor que *Pole Position*. El circuito es más grande y detallado, tiene balizas señalizadoras cada 100 metros y vallas que anuncian las curvas. También los coches contrarios están mejor definidos. Además, las explicaciones que aparecen en la parte superior de la pantalla son más interesantes: quién tienes delante, quien tienes detrás, el puesto que ocupas en la carrera y las vueltas que te faltan para acabar.

El grado de realismo es tal que se hace muy difícil manejar el bólido. Esto es un tanto a favor del programa: vale más un juego que comienza siendo difícil que otro que desde el principio se puede controlar con facilidad. En este sentido hay que decir que

## PROXIMAMENTE E

• ¿Puede haber algo mejor que jugar a algo contra un amigo en vez de contra el ordenador? Yo creo que no. De toda la vida los juegos han ido apareciendo con versiones para dos jugadores, bien sea por turnos o bien los dos a la vez porque este sistema de juego tiene más aceptación.

Los primeros eran del tipo "play player 1", "play player 2" y servían para que dos amigos jugaran a la vez y no discutieran. Pero aquello eran otros tiempos. Hoy en día, con los ordenadores personales, ese tipo de juegos ha pasado a la historia y, aunque la mayoría de los programas incluyen esa opción, casi nadie la utiliza.

Son los juegos en los que dos personas pueden participar simultáneamente (juegos bi-personales) los que tienen mayor aceptación. Son mis programas favoritos y lo que es más, los de muchas personas que conozco. Competir contra un ser humano en vez de contra una fría máquina —aunque la fría máquina sea nuestro querido 64— es siempre más agradable. Jugar al fútbol o al tenis contra el ordenador a veces se convierte en "misión imposible" y pierde uno el saborcillo del juego.

• Uno de los primeros juegos de este tipo fue *International Soccer* (Commodore International/distribuido por MEC). Los gráficos dejaban algo que desear —hacia poco que había salido a la calle el C-64—, pero los movimientos y el efecto tridimensional estaban, y están, muy bien. Jugar contra los amigos un pequeño torneo de fútbol puede ser algo apasionante, sobre todo si el nivel de juego es más o menos el mismo en todos. Si unos tienen más práctica o se saben todos los trucos habidos y por haber, el juego puede llegar a ser aburrido, aunque menos que cuando el rival es el ordenador. Hace poco (estas navidades) ha salido a la venta *International Basketball*, del mismo autor. Es un juego muy parecido al fútbol.

• Los juegos bi-personales casi siempre son de tipo deportivo. Ahí está el mítico *One-on-One* (Electronics Arts/Dro soft), un juego indispensable en la programotéca de cualquier "Commodoriano". Es una de las mejores simulaciones, tanto por los gráficos y los movimientos como por la competitividad existente.

Otros juegos de deporte para dos jugadores son *Starleague Baseball*, de Gamestar, un juego de béisbol, muy parecido al baseball de Epyx, pero a años luz de *Hardball* (Accolade/Compulogical) del cual tenéis la reseña en este mismo número. Los juegos de fútbol americano (rugby americano) son muy populares en Estados Unidos, pero aquí en España no hay ninguno todavía.

• *Pitstop II* (Epyx/Compulogical), sucesor del clásico *Pole Position* de Atari ha hecho también las delicias de los jugadores. Fue uno de los primeros programas que utilizaba la "split-screen" o "pantalla partida" para dar realismo al juego. En teoría consiste en que la pantalla queda dividida en dos mitades, una superior y otra inferior. Cada jugador ha de fijarse tan sólo en su pantalla, pues es en la que se ve a sí mismo frente a los demás coches, enemigos o lo que corresponda. Este sistema de presentación en pantalla surgió como solución a problemas que presentaban, por ejemplo, los *Summer Games* (Epyx/Compulogical) en el que los dos jugadores corrían en la misma pista. Si uno era muy bueno y corría mucho (o simplemente se jugaba contra el ordenador) el otro... ¡desaparecía de la pantalla! y no tenía más remedio que correr a ciegas.

Esta técnica la utilizan juegos como *Summer Games II* y *Winter Games* (Epyx/Compulogical), *Ballblazer* (Lucasfilm), *Spy vs Spy* (First Star), y muchos más. La forma más común de conseguir una pantalla partida es recurrir a las interrupciones por barrido de pantalla, una forma de

## N SUS PANTALLAS

programación que se tratará próximamente en Commodore World y que ofrece entre otras cosas la posibilidad de utilizar más de ocho sprites a la vez.

- En la gran mayoría de estos juegos los participantes juegan uno **contra** otro. Un buen ejemplo es **Spy vs Spy**, en el que como su propio nombre indica un espía juega contra otro. Este juego es muy entretenido y está basado en el comic-book americano **MAD**, una revista muy popular en Estados Unidos. En síntesis consiste en encontrar unos planos secretos que están escondidos en un edificio y luego escapar en una avioneta. Hace poco ha aparecido la segunda parte: **Spy vs Spy II, the island caper** en la que la aventura se desarrolla en una isla desierta.

- De momento, sólo un juego permite que dos personas actúen unidas: **Realm of impossibility** (Electronics Arts/Dro Soft) del cual tenéis la reseña en la revista del mes pasado. Este juego es muy variado e interesante y, aunque la temática es bastante clásica, la originalidad hacen de él algo excepcional.

- Todos los juegos bi-personales ofrecen opciones para jugar contra el ordenador. No es lo mismo jugar contra el ordenador en un juego de acción (cualquiera de marcianitos) en el que el ordenador actúa como "malo" utilizando todo lo que tiene a su disposición, que jugar contra un ordenador que controla un solo elemento —baloncestista, coche, boxeador— con las mismas características que las de nuestro personaje. Esto implica jugar contra el alguien determinado, no contra el "ambiente". Además, cuando se trata de un juego de marcianitos no hay "objetivo" que conseguir. Se trata sólo de resistir todo lo posible.

- Entre los juegos bi-personales que mejor hacen su papel al imitar a los humanos como jugadores están **One-on-One**, **Ballblazer** y **Soccer International**. Estos programas hacen que uno no se sienta impotente al utilizar al ordenador como "sparring" sino que más bien sirven para enseñar tácticas y como entrenamiento. Casi todos (por no decir todos) los juegos bi-personales aceptan diferentes niveles de dificultad. En **Ballblazer**, por ejemplo, estos niveles van de 1 a 9 y no se nota ningún cambio de velocidad, que es como suele determinarse el grado de dificultad del juego. Sin embargo **Tournament Tennis**, de Imagic, basa los niveles de dificultad en la velocidad, no sólo de la pelota, sino también de los jugadores. Cambiar la dificultad debe ser algo más que simplemente variar la velocidad. El grado de "inteligencia" ha de consistir en planificación, técnicas y "aprendizaje" por parte del ordenador lo cual, de momento, sólo lo presentan algunos programas de ajedrez, en los que no se puede decir que haya mucha acción.

Los programas que "peor" juegan (todo es relativo) son **Pitstop II** y **Gamestar Baseball**. Es relativamente fácil ganar al coche de **Pitstop II**, porque el nivel de habilidad no se basa en la "inteligencia" de los conductores sino en los golpes que pueden sufrir —además de que el coche conducido por el ordenador no alcanza nunca la velocidad máxima—. **A Gamestar Baseball** no hay quien le gane, el ordenador deja que sus jugadores vayan más rápidos que los demás, y eso desmoraliza a cualquiera.

- Se quedan muchos juegos de este tipo en el tintero: **Racing Destruction Set** (Electronics Arts/Dro Soft), del que hablaremos el mes que viene, **M.u.l.e.** en el que pueden participar hasta cuatro jugadores —pero no del todo simultáneamente— y muchos otros que no se conocen en España todavía. Poco a poco irán viniendo y os los anunciaremos con gusto.

aquí no hemos conseguido todavía completar una vuelta al circuito (bueno, una sí, pero en primera). Cada vez que el coche se estrella hay que volver a empezar. Esto da un poco de "interés" al juego. Lo que no puede ser es, como en otros juegos, que cuando el coche se choca, estalle en una magnífica explosión de colores y sonido para luego volver a aparecer tan campante como si nada hubiera sucedido.

## Diccionario para su Commodore 64

**Autores:** Thomas Jordan  
y Wolfgang Schellenberger.  
**Traductora:** Petra Scheller.  
**Edición original:** Data Becker.  
**Edición española:** Ferré Moret.  
**Tuset, 8 entlo. 2.**  
**08006 Barcelona.**  
**Teléf.:** (93) 218 02 93.  
**Páginas:** 320.  
**Precio:** 2.800 ptas.

**T**al y como nos lo podemos imaginar, este libro es un diccionario técnico en torno al mundo de los micros y concretamente el Commodore 64.

La idea desde luego es muy interesante, ya que no conozco ningún libro de estas características. Además los diccionarios técnicos (aceptables) de informática a este nivel no abundan mucho en España (pero si existen algunas recopilaciones, como la sección de glosario de esta revista). Esto nos lleva al problema del lenguaje técnico en informática, en que la mezcla de vaguedad y orgullo nacional nos han llevado a un argot propio, que ni es inglés, ni es español. Bueno, no es cosa de discutir la cuestión en este lugar y momento.

Este libro, para salir airoso del paso sin mancharse mucho, y con el prestigio y seriedad de la conocida Data Becker, no se limita a darnos una relación de las palabras técnicas que nos podamos encontrar en el argot de por aquí, sino que nos da además otra lista de los términos equivalentes (no me atrevo a utilizar el término "traducidos") en el argot anglo-americano de los ordenadores personales.

El tipo de cosas que incluye son los comandos del Basic del 64, las diferentes características del ordenador, sus posibilidades, las de sus periféricos, y en general cualquier palabra que pudiera aparecer en relación con la utilización de este ordenador.

En muchas de las palabras, las definiciones vienen explicadas con ejemplos, tablas o lo que haga falta. A pesar de todo, yo no creo que el libro pretenda ser un libro didáctico, en el sentido de que no es un manual de





aprendizaje, aunque si se puede aprender mucho de él.

Como ya he dicho antes, además de todos

los conceptos propios del Commodore 64, podemos encontrar en el libro una multitud de términos informáticos de todo tipo, que uno se puede encontrar en cualquier momento si está trabajando con ordenadores, desde nombres de lenguajes hasta tipos de periféricos, palabras de Basic Microsoft, términos de programación general, etcétera.

Como todo buen diccionario que se respete, este libro está estructurado por estricto orden alfabético, excepto las listas o tablas que están dentro de una definición. Quizá esto sea una importante ventaja sobre otros libros de referencia, sobre todo, a la hora de encontrar información sobre una cosa en concreto.

Ahora bien, la principal objeción que se le puede poner a este libro es su nivel. Desde luego no se le puede tratar de enciclopedia a un libro de 320 páginas, tamaño cuartilla, por muy pequeña que sea la letra. A pesar de que el campo que el libro abarca sea bastante amplio, personalmente, yo no creo que se

extienda en la dirección adecuada. Los términos del Commodore 64 en sí son bien sencillos, muchos de ellos se pueden encontrar más extensivamente y mejor explicados en el mismo manual del usuario, o en cualquier guía del programador si me apuráis mucho. En cuanto a esos términos raros de informática que aparecen en el libro, la verdad es que a nivel de usuario de Commodore 64, no aparecen casi nunca, e incluso a niveles superiores, más que otra cosa porque han caído en desuso.

De todas maneras, ahí queda la ventaja de tenerlo todo reunido en un solo libro. Yo creo que también puede ser interesante para un novato en este mundillo, que además de conocer su recién adquirido ordenador, quiere saber de qué va la cosa un poquito más allá de su Commodore 64, por curiosidad. También les podría venir bien a algunos traductores técnicos que no entiendan mucho del tema, para que por fin sepan qué cosas se deben traducir literalmente y cuáles no.

## Juegos y programas educativos para C-64

*Autor: George Stewart.  
Traductores: Antonio Lloris Ruiz  
y Alberto Prieto Espinosa.  
Editor:  
Mc Graw-Hill de España.  
Santa Beatriz, 4.  
28018 Madrid.  
Teléf.: 433 87 77/78/79.  
Páginas: 320.  
Precio: 1.420 ptas.*

**E**ste parecía ser uno de tantos libros que aparecen por allí, en que se nos ofrece por el precio del libro una decena de juegos "excitantes", de aventura, de acción, de ingenio, etc., pero con el gran inconveniente de que están en Basic (esto no significa que no se puedan hacer cosas interesantísimas en Basic), y no tiene nada de acción, aventura o ingenio, y para que teclear programas están los de Commodore World que suelen ser bastante mejores.

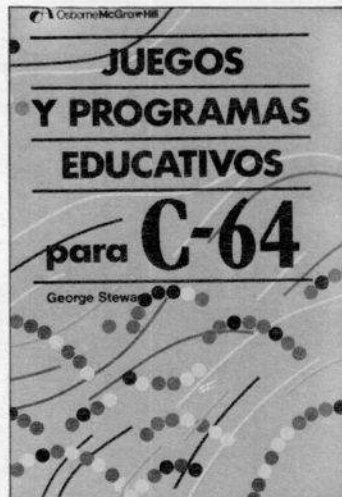
Pero generalmente, y este libro en concreto, estos libros presentan otros aspectos más interesantes que el juego en sí. Muchos de estos programas de juegos vienen ampliamente comentados, explicando las técnicas de programación que se han utilizado para realizarlos. Así si no lo sabes ya, en este libro te vienen muy buenas ideas de cómo programar un laberinto, programar un crucigrama en el ordenador, crear un analizador de textos y muchas ideas más (20 programas en total).

A medida que iba leyendo el libro, y realizando algunos de sus programas, me convencí de lo que aquí menos importaba era la calidad de los programas. Ante todo este libro es una fuente de ideas originales, desde

las ideas de los programas en sí hasta los pequeños detalles de programación, que a mí nunca se me habrían ocurrido, no muy importantes al final, pero que son realmente ingeniosos. Pero si he aprendido algo con este libro, no se trata de técnicas de programación ni de cómo hacer virguerías con el ordenador. Tal y como lo dice el título, este es un libro didáctico, pero no hay que interpretarlo creyendo que al final vamos a acabar siendo unos fieras en programación.

Este libro es un ejemplo de lo que se puede hacer con los ordenadores en el campo de la educación, y no hace falta que sean bien grandes ni que tengan muchos periféricos ni nada de eso. Esto demuestra cómo el uso de los ordenadores en la educación no tiene por qué limitarse al campo de las matemáticas o de cualquier otra ciencia exacta. Dos ejemplos en este libro: un programa para hacer acertijos que es una auténtica lección de lógica de clases, y otros programa, bien sencillo, que analiza un texto contando las palabras y letras que tiene, dando sobre el texto unos valores estadísticos de mucho interés; ahí está, dos aplicaciones en dos asignaturas que en principio tienen muy poco que ver con la informática, la filosofía y la lengua.

Además aquí se desarrolla una parte de la programación, se trata de la resolución del algoritmo en sí (hablando en cristiano, cómo demonios hago que el ordenador dibuje aleatoriamente y luego almacene el laberinto de mi juego). El ejemplo del laberinto es muy típico. De lo que aquí se trata es de fomentar el razonamiento lógico del niño (bueno, bueno, el joven estudiante-programador) que está intentando hacer un programa, sin más armas que sus conocimientos de Basic.



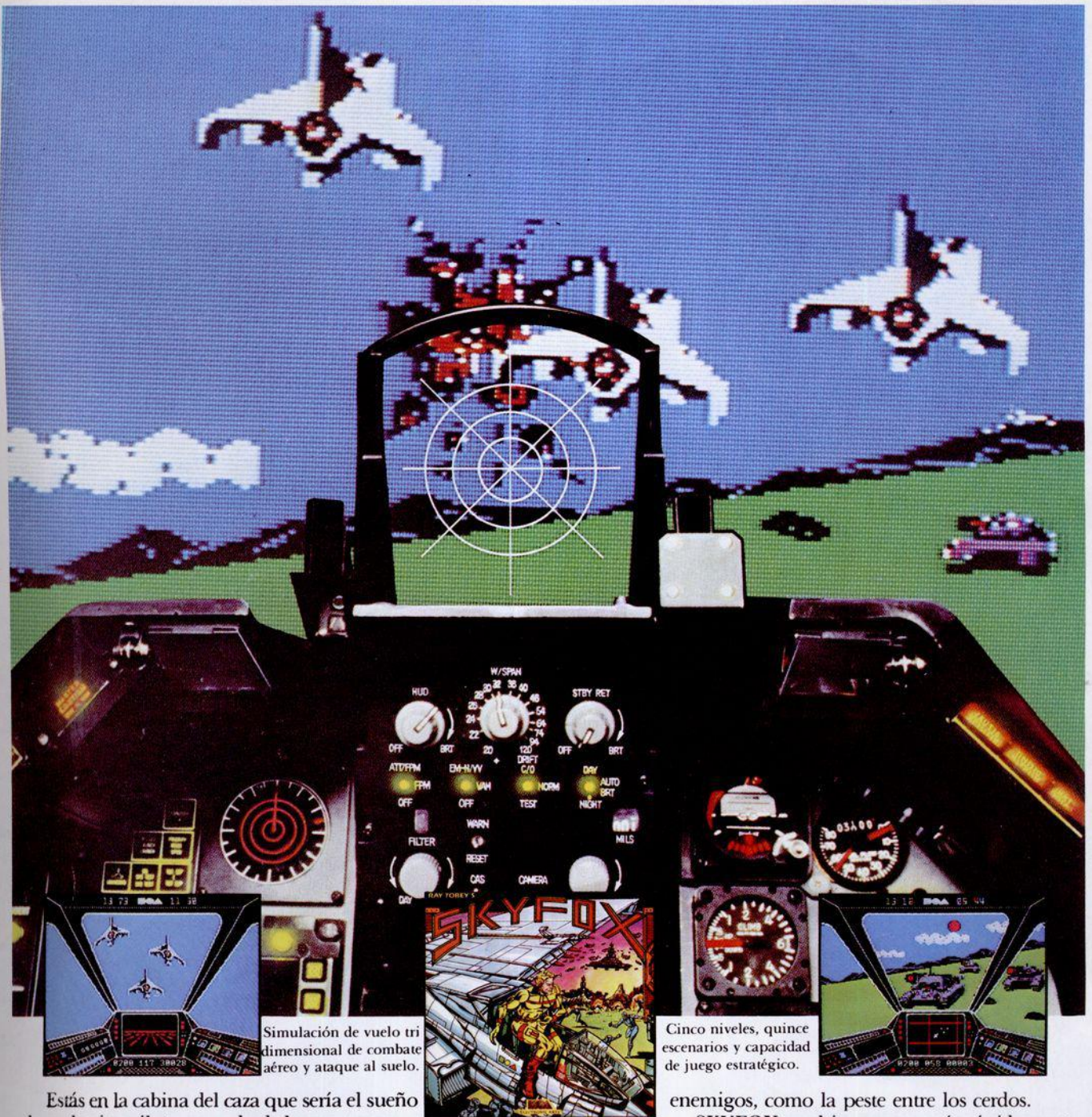
Ojalá sigan saliendo libros como estos para que poco a poco vayamos comiéndole el coco al personal docente para que nuestros tataranietos no tengan que seguir aprendiendo con métodos medievales e improductivos como nosotros lo hicimos.

**Avisamos a  
todos los suscriptores  
que nuestro  
departamento  
de suscripciones  
ha cambiado  
de teléfonos:**

**221 86 71  
(91) 221 86 77**



# Defiende tu espacio aéreo



Simulación de vuelo tridimensional de combate aéreo y ataque al suelo.

Cinco niveles, quince escenarios y capacidad de juego estratégico.

Estás en la cabina del caza que sería el sueño de cualquier piloto, pero desde luego eres un mal sueño para el pobre tipo que tienes delante, confiado en una misión sin problemas. Calientale la tobera con tus láser y apartate mientras estalla en una bola de fuego. Rápidamente ponte en picado para caer sobre los blindados

enemigos, como la peste entre los cerdos.

SKYFOX es el juego que más rápidamente se está vendiendo en toda la historia de Electronic ARTS.

Posee la más asombrosa animación de alta velocidad que hayas visto en tu ordenador.

Ahora puede ser tuyo totalmente traducido al castellano.

# SKYFOX™

## EN CASTELLANO

P.V.P. 2.500 ptas.



**DRO SOFT**



**ELECTRONIC ARTS**

**CARACTERISTICAS:** NACIONALIDAD: Federación galáctica. **FABRICANTE:** TOBEY ASTRONAUTICS **TIPO:** Caza interceptor multipropósito **PROPULSION AUXILIAR:** Un generador antigraavitatorio a 66 MKI **TRIPULACION:** Un humano. **ARMAMENTO:** Dos cañones láser de fuego continuo de 70 kilojulios 10 toneladas de empuje. 5 misiles rastreadores de calor tipo PHOENIX 5 misiles guiados por radar tipo TYPHOON **DEFENSA:** — 2 unidades deflectoras WCRC **AYUDAS ELECTRONICAS:** Radar SCANNER de largo y corto alcance conectable al piloto automático. **VELOCIDAD EN ATMOSFERA:** — 3.000 MPH (Mach IV a 35.000 piés).

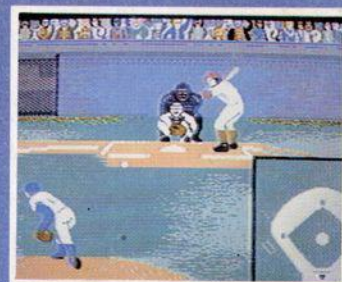
Editado por DRO SOFT Fundadores, 3 - 28028 Madrid.

Tel.: 255 45 00/09



# HardBall

Nunca verá un juego de béisbol  
tan próximo a la realidad



Esto es el **HARDBALL**, simple y a su vez el juego de simulación de deportes más realista de todos los tiempos. Le bastarán sólo cinco minutos para ver que todos los demás juegos de béisbol para ordenadores son de menor categoría en comparación con el **HARDBALL**. Podría jurar que está viendo un programa de la televisión un sábado por la tarde. Al **HARDBALL** se puede jugar de dos maneras, una como juego de acción en el campo, y otra como un juego de estrategia de entrenador, o ambas a la vez. Observe la curva descrita por la bola lanzada por encima de la rotunda o consulte la pantalla de entrenadores para una sustitución clave. Puede incluso situarse dentro o fuera del terreno de juego para comprobar el estilo del bateador o la situación del juego.